МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ-2»

КОМП'ЮТЕРНИЙ ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю 171 «Електроніка»,
освітньою програмою «Електронні компоненти і системи»*

Київ
КПІ ім. Ігоря Сікорського
2020

«Інформаційні технології-2»: комп'ютерний практикум [Електронний ресурс]: навч. посіб. для студ. спеціальності 171 «Електроніка», освітньої програми «Електронні компоненти і системи» / КПІ ім. Ігоря Сікорського ; уклад.: К. С. Клен. – Електронні текстові данні (1 файл: 3,7 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2020. – 147 с.

Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 5 від 14.01.2020 р.)
за поданням Вченої ради факультету електроніки (протокол № 12/2020 від 21.12.2020 р.)

Електронне мережне навчальне видання

«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ-2»
КОМП'ЮТЕРНИЙ ПРАКТИКУМ

Укладачі: *Клен Катерина Сергіївна,* канд. техн. наук, доц..

Відповідальний редактор *Ямненко Ю. С.*, завідувач кафедри електронних пристроїв та систем, д-р техн. наук, проф.

Рецензент: *Найда С.А.,* завідувач кафедри акустичних та мультимедійних електронних систем, д-р техн. наук, проф.

При підготовці бакалаврів за спеціальністю 171 Електроніка, освітньою програмою «Електронні системи» однією з важливих дисциплін, що є компонентом циклу професійної підготовки, є дисципліна «Інформаційні технології-2».

Метою розробки комп'ютерного практикуму є дати студентам грунтовні знання основних констант, операторів та функцій програми Mathematica, правил складання програм та призначень пакетів розширення програми Mathematica. В результаті вивчення матеріалу комп'ютерного практикуму студент повинен отримати уміння робити інженерні розрахунки; користуватись засобами візуалізації, інтерфейсу користувача і побудови графіків; складати програми обчислень.

Комп'ютерний практикум містить теоретичні відомості та завдання до 8 лабораторних робіт та список рекомендованої літератури.

ЗМІСТ

Вступ

Програмний пакет Mathematica є системою комп'ютерної алгебри, призначеною для обробки математичних формул. Серед інших систем комп'ютерної алгебри слід відзначити програми Maxima, Maple, MuPAD, Reduce. Основна задача названих програм є автоматизація алгебраїчних перетворень і обробки символьних виразів. Перша версія пакету Mathematica розроблена у 1988 році, версія 11.0.1 пакету вийшла у вересні 2016 року.

У комп'ютерному практикумі розглянуто основні можливості системи Mathematica з обробки масивів даних, побудови графіків, розв'язку лінійних і трансцендетних рівнянь, розв'язку задач математичного аналізу, апроксимації даних, основ програмування і створення інтерфейсу користувача.

Лабораторна робота № 1. Основи роботи з пакетом Mathematica

Мета: вивчення базових можливостей системи Mathematica.

Короткі теоретичні відомості

Основні можливості системи Mathematica:

- інтегрування і диференціювання функцій, вирішення систем поліноміальних і тригонометричних рівнянь та нерівностей, рекурентних рівнянь, рішення диференційних рівнянь і рівнянь в частинних похідних, ряди Тейлора, спрощення виразів, розрахунок меж, знаходження скінченних і нескінченних сум і добутків, а також ряд інших задач у символьному вигляді;
- поліноміальна інтерполяція функцій, розрахунок елементарних і спеціальних функцій із заданим ступенем точності, розрахунок перетворень Лапласа, Фур'є, z-перетворення;
- розв'язок задач лінійної алгебри, теорії чисел та інших розділів математики;
- представлення даних у графічному форматі (побудова графіків, параметричних кривих і поверхонь, побудова геометричних фігур, імпорт і експорт графічних даних у растрових і векторних форматах);
- підтримка розподілених обчислень (пакет Parallel Computing Toolkit);
- можливість написання програм на вбудованій процедурно-функціональній мові програмування.

Крім того, система має ряд стандартних пакетів розширення (Add-Ons):

- Algebra – робота з поліномами, алгебраїчними нерівностями, Гамільтоновою алгеброю і т.д.;
- Calculus – символьні обчислення похідних, інтегралів меж функцій, пряме і зворотне перетворення Фур'є і Лапласа, вирішення систем нелінійних рівнянь, реалізація інваріантних методів, вирішення диференційних рівнянь у частинних похідних, знаходження повних інтегралів і диференційних інваріантів нелінійних рівнянь, апроксимація Паде, обчислення еліптичних інтегралів і робота з векторами;
- DiscreteMath – обчислення з області дискретної математики, комбінаторики, обчислювальної геометрії і теорії графів, вирішення рекурентних і різницевих рівнянь, операції з цілими числами і т.д.;
- Geometry – функції для виконання геометричних розрахунків, створення правильних прямокутників і багатогранників, обертання геометричних фігур на площині і в просторі;
- Graphics – побудова графіків спеціального виду, геометричних фігур і поверхонь, графіків параметрично і неявно заданих функцій, опис функція комплексної змінної , відображення ортогональних проекцій тривимірних фігур, імітація тіней, засоби оформлення графіків;
- LinearAlgebra – вирішення задач лінійної алгебри, додаткові векторні і матричні операції, формування ортогональних векторних базисів;
- Miscellaneuos – визначення одиниць вимірювання фізичних величин, даних про хімічні елементи, фізичні константи, географічні дані та ін.;

- NumberTheory – функції теорії чисел;
- NumericalMath – реалізація чисельних методів, апроксимація даних і аналітичних функцій поліномами, сплайнами, тригонометричними рядами, чисельне інтегрування і диференціювання, вирішення диференційних рівнянь, обчислення коренів нелінійних рівнянь;
- Statistics – статистичні функції для неперервних і дискретних функцій розподілу, реалізація лінійної і нелінійної регресії, обчислення параметрів функцій розподілу;
- Utilities – додаткові утиліти для роботи з бінарними файлами і пам’яттю комп’ютера, підтримки мов, роботи з системами класу AutoCAD і т.д.

Основи роботи з системою

Інтерфейс системи Mathematica складається з головного меню і області введення даних – блокноту, рис. 1.

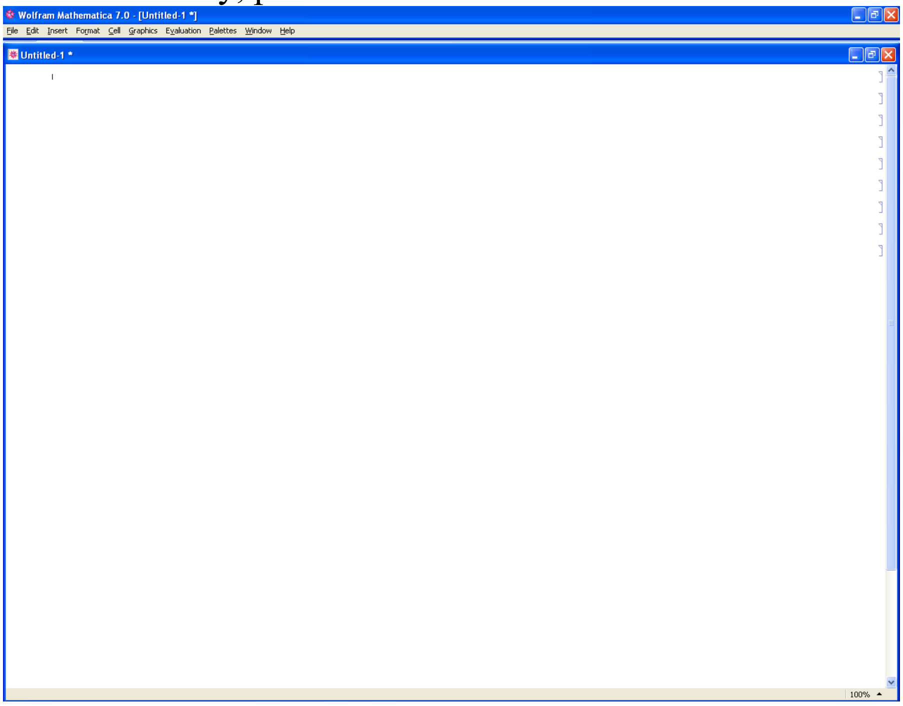


Рис. 1. Інтерфейс системи Mathematica

Робочим документом системи Mathematica є блокнот. Блокнот складається з комірок. Найпростіший спосіб роботи з системою Mathematica – інтерактивний. Система призначає кожному введеному і виведеному виразу номер комірки – In[n] і Out[n] відповідно. Для використання останнього виразу достатньо ввести «%», для посилання на результат, записаний в комірці під номером n – «%n». Для виконання введеної команди в кінці рядка необхідно натиснути клавіші Shift+Enter. Результатом виконання математичних операцій над виразами, вказаними у лістингу 1, будуть вирази, записані у вихідних комірках Out[1]-Out[6], лістинг 2.

```
5^10
4+5
%+a
%2
4+5
9
```

Лістинг 1

```
Out[1]= 9765625
Out[2]= 9
Out[3]= 9+a
Out[4]= 9
Out[5]= 9
Out[6]= 9
```

Лістинг 2

При введенні даних необхідно дотримуватись наступних правил:

- квадратні дужки [] використовуються для позначення аргументів функцій, навіть якщо функція не має аргументів, наприклад Random[];
- фігурні дужки використовуються для створення списків, векторів і матриць;
- круглі дужки використовуються в математичних виразах для задавання пріоритету виконання математичних операцій;
- подвійні квадратні дужки використовуються для індексації елементу у списку. [[i]] – повертає і-й елемент списку, [[i,j]] – повертає j-й елемент в рядку і.

Оперативна довідка

В системі Mathematica наявна коротка довідка по використовуваним в її середовищі об’єктам. Для виводу всього переліку об’єктів необхідно ввести команду:

?*

Можна також отримати довідку по всім об’єктам, назва яких починається на певну літеру:

?U*

Для отримання короткої довідки по конкретному об’єкту необхідно ввести команду ?Name, наприклад:

?Abs

Abs[z] gives the absolute
value of the real or complex number z.

Арифметичні операції

Операція	Арифметичний оператор	Скорочена форма	Функція
Додавання	+	+=	Plus[x1, x2,…xn]
Віднімання	-	-=	-
Множення	*	*=	Times[x1, x2,…xn]
Ділення	/	/=	Divide[x1, x2]
Піднесення до степеня	^	-	-

Типи даних

Тип чисел	Позначення	Приклад
Цілочисельні	Integer	35, -2
Раціональні	Rational	35/46,
Дійсні	Real	2.18, 3.6*10^-5
Комплексні	Complex	2+3*I

Іменовані константи

E – число е;

Pi – число π;

I – уявна одиниця $\sqrt{-1}$;

Infinity – уявна нескінченість $+\infty$, при від’ємній нескінченості ставиться знак -;

Degree – число радіан в градусі π/180;

EulerGamma – постійна Ейлера 0,577216;

GoldenRatio – константа золотого перерізу $\frac{1+\sqrt{5}}{2}$;

Catalan – константа Каталана 0.915966.

Для отримання чисельного значення константи необхідно застосувати функцію N[].

N[E]

Out[1]= 2.71828

Якщо необхідно вивести значення числа із заданою кількістю знаків, використовується функція N[k,n], де k – число, n – кількість знаків.

N[5/88,10]

Out[1]= 0.05681818182

Вбудовані математичні функції

Функції визначення дільників, найменшого кратного цілих чисел

Divisors[n] – повертає цілочисельні дільники числа n;

ExtendedGCD[n,m] – повертає найбілший спільний дільник чисел n і m;

GCD[n1,n2…] – повертає найбільший спільний дільник чисел n1, n2…;

LCM[n1,n2…] – повертає найменше спільне кратне чисел n1, n2…

Mod[x,y] – повертає залишок від ділення числа x на y.

Функції округлення дійсних чисел

Round[x] – округлення до найближчого цілого;

Floor[x] – повертає найбільше число, що не перевищує x;

Ceiling[x] – повертає найменше число, що більше або дорівнює x;

Quotient[x,y] – повертає округлене ціле число x/y, що не перевищує значення x/y.

Обчислення факторіалів

Factorial[n] – повертає значення n!;

Factorial2[n] – повертає значення n!!=n*(n-2)*(n-4)...

Отримання простих чисел

Prime[n] – повертає n-е просте число;

PrimePi[n] – повертає кількість простих чисел, що не перевищують n.

Елементарні функції

Степеневі і логарифмічні функції

$\sqrt{z} \to$ Sqrt[z];

$z^a \to$ Power[z,a];

$e^z \to$ Exp[z];

$\ln(z) \to$ Log[z];

$\log_a(z) \to$ Log[a,z];

Тригонометричні функції

$\sin(z) \to$ Sin[z];

$\cos(z) \to$ Cos[z];

$tg(z) \to$ Tan[z];

$ctg(z) \rightarrow$ Cot[z];

$\csc(z) \rightarrow$ Csc[z];

$sec(z) \rightarrow$ Sec[z].

Обернені тригонометричні функції

$arc\sin(z) \rightarrow$ ArcSin[z];

$arc\cos(z) \rightarrow$ ArcCos[z];

$arcctg(z) \rightarrow$ ArcCot[z];

$arc\csc(z) \rightarrow$ ArcCsc[z];

$arcsec(z) \rightarrow$ ArcSec[z].

Гіперболічні функції

$\sinh(z) \rightarrow$ Sinh[z];

$\cosh(z) \rightarrow$ Cosh[z];

$tgh(z) \rightarrow$ Tanh[z];

$ctgh(z) \rightarrow$ Coth[z];

$\csc h(z) \rightarrow$ Csch[z];

$\sec h(z) \rightarrow$ Sech[z].

Обернені гіперболічні функції

$arc\sinh(z) \rightarrow$ ArcSinh[z];

$arc\cosh(z) \rightarrow$ ArcCosh[z];

$arcctgh(z) \rightarrow$ ArcCoth[z];

$arc\csc h(z) \rightarrow$ ArcCsch[z];

$arc\sec h(z) \rightarrow$ ArcSech[z].

Арифметичні операції з цілими і раціональними числами

Система Mathematica виконує обчислення з цілими і раціональними числами без похибок, що ілюструється в лістингу 3.

```
In[1]:= 100 !!

Out[1]= 34 243 224 702 511 976 248 246 432 895 208 185 975
         118 675 053 719 198 827 915 654 463 488 000 000
         000 000

In[2]:= 1 / 2 + 2 / 3 + 1 / 6

Out[2]= 4/3

In[3]:= 2 / 17 + 3 / 7 + 5 / 22 + 1 / 121

Out[3]= 22 513/28 798
```

Лістинг 3

Арифметичні операції з дійсними числами

Дійсні числа в системі Mathematica представляються в звичайній або нормальній формі. При представленні числа в звичайній формі ціла частина числа відділяється від дробовою крапкою: 1.35, 0.24. При цьому 0 цілих можна не писати і замість 0.25 використовувати запис - .25. Числа записані в такій формі запису називаються числа з фіксованою крапкою.

Крапка в кінці числа є ознакою, що число є дійсним. Наприклад число 131 є цілим, число 131. – дійсним, число 2/5 – є раціональним, а число 2./5 – дійсним.

При представленні числа в нормальній формі, число записується у вигляді мантиси з цілою і дробовою частиною і порядку у вигляді степеня числа: 5.*10^-3, 3.335*10^-6. Замість знаку множення можна використовувати пробіл. Арифметичні операції над дійсними числами дають наближений результат. Система Mathematica оперує числами в діапазоні $2.22507^{-308}...1.79769\cdot 10^{308}$. Для підвищення точності дійсні числа можна переводити в раціональні за допомогою функцій:

Rationalize[z] – дійсне число z перетворюється в раціональне;

Rationalize[z,dz] – дійсне число z перетворюється в раціональне з точністю dz;

Арифметичні операції з комплексними числами

Комплексне число представляється в такому виді:

$z = \mathrm{Re}(z) + I * \mathrm{Im}(z)$.

Функції виконання операцій над комплексними числами:

Abs[z] – обчислює модуль комплексного числа z;

Arg[z] – обчислює аргумент комплексного числа z;

Conjugate[z] – обчислює комплексноспряжене число до z;

Вирази їх перетворення і обчислення

Підстановки

Підстановки є математичним апаратом, призначеним для обчислення функцій при чисельно заданих значеннях аргументу. Вони дозволяють табулювати значення функцій. В системі Mathematica для цього використовується символ «/.»

f(x) /. x->a;

f(x,y,…) /. {x->a,y->b,…};

{f1(x),f2(x),…} /. x->a;

{f1(x,y,..),f2(x,y,…),…} /. {x->a,y->b,…};

f(x) /. x->{x0,x1,…}.

Вирази підстановок мають наступний зміст:

f(x) /. x->a – здійснює підстановку в вираз f(x) значення x=a.

f(x,y,…) /. {x->a,y->b,…} – здійснює підстановку в вираз f(x,y,…) значення x=a, y=b,…

{f1(x),f2(x),…} /. x->a – здійснює підстановку в вирази f1(x), f2(x),… значення x=a.

{f1(x,y,..),f2(x,y,…),…} /. {x->a,y->b,…} – здійснює підстановку в вирази f1(x,y,…), f2(x,y,…),… значення x=a, y=b,…

f(x) /. x->{x0,x1,…} – табулювання функції f(x).

Приклад використання підстановок показано у лістингу 4.

```
In[11]:= z1 = x * E^x + Log[x] - 1;
         z1 /. x → 1

Out[12]= -1 + ⅇ

In[13]:= z2 = x^2 + y^2 + Sin[x]^2 + Cos[x]^2 + 1.5;
         z2 /. {x → 1, y → 2}

Out[14]= 7.5

In[15]:= z2 /. y → 2

Out[15]= 5.5 + x^2 + Cos[x]^2 + Sin[x]^2

In[16]:= z1 /. x → a

Out[16]= -1 + a ⅇ^a + Log[a]

In[17]:= z2 /. {x → a, y → b}

Out[17]= 1.5 + a^2 + b^2 + Cos[a]^2 + Sin[a]^2

In[18]:= z2 /. {x → a, y → a}

Out[18]= 1.5 + 2 a^2 + Cos[a]^2 + Sin[a]^2
```

Лістинг 4

Перетворення виразів

Функцій перетворення виразів:

Simplify[f] – спрощує вираз f;

FullSimplify[f] - спрощує вираз f, який в своєму складі містить спеціальні функції;

Expand[f] – розкриває і розширює вираз f;

Collect[f,x] – представляє вираз f по степеням x;

TrigExpand[f] – перетворює тригонометричні вирази;

Factor[f] – розкладає вираз на множники.

Приклади використання функції Simplify показано в лістингу 5, FullSimplify – в лістингу 6.

```
In[20]:= Simplify[957828 / 3831312]
```

Out[20]= $\frac{1}{4}$

```
In[21]:= Simplify[(315 + 297 x - 62 x^2 - 42 x^3 + 3 x^4 + x^5) / (105 - 41 x - x^2 + x^3)]
```

Out[21]= $3 + 4\,x + x^2$

```
In[22]:= Simplify[(Sin[x] + Cos[x])^2 - 1]
```

Out[22]= Sin[2 x]

```
In[23]:= Simplify[a^4 + (a^3) x + (a^2) x^2 + a (x^3) + x^4 + a (-1 + a^4) / (x - a)]
```

Out[23]= $\frac{a - x^5}{a - x}$

```
In[28]:= Simplify[7.59375 + 25.3125 x + 33.75 x^2 + 22.5 x^3 + 7.5 x^4 + x^5]
```

Out[28]= $1.\,(1.5 + 1.\,x)^5$

```
In[29]:= Simplify[(2 x + a^Log[a, x] + 3) / 3]
```

Out[29]= 1 + x

Лістинг 5

```
In[30]:= Simplify[x * Sin[x] * Cos[y] + (x^2 + 1) / x * Sin[y] * Cos[x]]
```

Out[30]= $x\,\mathrm{Cos}[y]\,\mathrm{Sin}[x] + \frac{(1 + x^2)\,\mathrm{Cos}[x]\,\mathrm{Sin}[y]}{x}$

```
In[31]:= FullSimplify[x * Sin[x] * Cos[y] + (x^2 + 1) / x * Sin[y] * Cos[x]]
```

Out[31]= $\frac{\mathrm{Cos}[x]\,\mathrm{Sin}[y]}{x} + x\,\mathrm{Sin}[x + y]$

```
In[32]:= Simplify[(Sin[x] + Cos[x])^2]
```

Out[32]= $(\mathrm{Cos}[x] + \mathrm{Sin}[x])^2$

```
In[33]:= FullSimplify[(Sin[x] + Cos[x])^2]
```

Out[33]= 1 + Sin[2 x]

```
In[34]:= FullSimplify[(1 - Cos[2 x]) / 2 + Cos[x]^2 + a]
```

Out[34]= 1 + a

Лістинг 6

Функція Expand

Модифікації функції Expand:

Expand[f] – розкриває дужки добутків і підносить вирази в цілий додатний степінь;

ExpandAll[f] – розкриває дужки добутків і підносить вирази в цілий додатний степінь у будь-якому місці виразу f;

ExpandNumerator[f] - розкриває дужки добутків і підносить вирази в цілий додатний степінь у чисельнику виразу f;

ExpandDenominator[f] - розкриває дужки добутків і підносить вирази в цілий додатний степінь у знаменнику виразу f;

PowerExpand[f] - розкриває дужки добутків і підносить вирази в цілий додатний степінь у вкладених виразах функції f;

ComplexExpand[f] - розкриває дужки добутків і підносить вирази в цілий додатний степінь, за умови що змінні виразу f є комплексними числами;

ComplexExpand[f,{x1,x2,…}] - розкриває дужки добутків і підносить вирази в цілий додатний степінь, за умови що змінні x1,x2,.. виразу f є комплексними числами;

FunctionExpand[f] – використовується для спрощення виразу f, якщо до його складу входять спеціальні функції.

В лістингу 7 показані приклади використання модифікацій функції Expand[].

```
In[35]:=
Expand[(a + b) (a - b) (a + c) / (a + 1) ^3]
```

Out[35]= $\frac{a^3}{(1+a)^3}-\frac{a\,b^2}{(1+a)^3}+\frac{a^2\,c}{(1+a)^3}-\frac{b^2\,c}{(1+a)^3}$

```
In[36]:= ExpandAll[(a + b) (a - b) (a + c) / (a + 1) ^3]
```

Out[36]= $\frac{a^3}{1+3\,a+3\,a^2+a^3}-\frac{a\,b^2}{1+3\,a+3\,a^2+a^3}+\frac{a^2\,c}{1+3\,a+3\,a^2+a^3}-\frac{b^2\,c}{1+3\,a+3\,a^2+a^3}$

```
In[37]:= ExpandAll[Sqrt[x + 2^3] / ((a + b) (a - b)) (a - b) + (c + 2 I) ^2]
```

Out[37]= $-4+4\,i\,c+c^2+\frac{\sqrt{8+x}}{a+b}$

```
In[38]:= ExpandNumerator[(x^2 - 1) ^5 / (x + 1) ^3]
```

Out[38]= $\frac{-1+5\,x^2-10\,x^4+10\,x^6-5\,x^8+x^{10}}{(1+x)^3}$

```
In[39]:= ExpandDenominator[(x^2 - 1) ^5 / (x + 1) ^3]
```

Out[39]= $\frac{\left(-1+x^2\right)^5}{1+3\,x+3\,x^2+x^3}$

Лістинг 7

Функція Collect

Модифікації функції Collect:

Collect[f,x] – представляє вираз f по степеням x;

Collect[f,{x1,x2,…}] – представляє вираз f по степеням змінних x1,x2...

В лістингу 8 показані приклади використання модифікацій функції Collect[].

```
In[40]:= Collect[(x + a)^3 - (x - a)^2, x]
```

Out[40]= $-a^2 + a^3 + (2\,a + 3\,a^2)\,x + (-1 + 3\,a)\,x^2 + x^3$

```
In[41]:= Collect[(x + 1)^5 - EulerE[5, x] - (x + a)^3, x]
```

Out[41]= $\frac{3}{2} - a^3 + (5 - 3\,a^2)\,x + \left(\frac{15}{2} - 3\,a\right)x^2 + 9\,x^3 + \frac{15\,x^4}{2}$

```
In[42]:= Collect[(x - y) (x + y)^2 Sqrt[(x + y)^3], x]
```

Out[42]= $x^3\sqrt{(x+y)^3} + x^2\,y\sqrt{(x+y)^3} - x\,y^2\sqrt{(x+y)^3} - y^3\sqrt{(x+y)^3}$

```
In[43]:= Collect[(x + y)^2 + (x - 1)^3 - (y - 1)^3, {x, y}]
```

Out[43]= $-2\,x^2 + x^3 - 3\,y + 4\,y^2 - y^3 + x\,(3 + 2\,y)$

```
In[44]:= Collect[1 / (x + y) + 1 / (x - y) + x - y, {x, y}]
```

Out[44]= $x + \frac{1}{x - y} - y + \frac{1}{x + y}$

Лістинг 8

Функція Factor

Модифікації функції Factor:

Factor[f] – розклад функції f на прості доданки;

FactorList[f] – повертає перелік множників і число степенів кожного з них;

FactorTerms[f] – виносить за дужки спільний множник виразу f;

FactorTermsList[f] – виділяє спільний множник виразу f, представляючи функцію f в іншому виді;

FactorInteger[f] – повертає прості множники цілого числа f.

В лістингу 9 показані приклади використання модифікацій функції Factor[].

```
In[45]:= Factor[18 x^3 + 45 x^2 + x - 14]
Out[45]= (-1 + 2 x) (2 + 3 x) (7 + 3 x)

In[46]:= Factor[x^3 + x^2 (3 + a - b) + x (3 a - 3 b - a * b) - 3 a * b]
Out[46]= - (b - x) (3 + x) (a + x)

In[47]:= Factor[a^2 + a - 1]
Out[47]= -1 + a + a^2

In[48]:= Factor[6 x^3 + x^2 (13 I - 6) - 6 x (1 + 13 / 6 I) + 6]
Out[48]= (-1 + x) (3 i + 2 x) (2 i + 3 x)

In[51]:= Factor[1 / 16 (Cos[5 x] + Cos[3 x] + 10 Cos[x]), Trig → True]
Out[51]= 1/8 Cos[x] (5 + Cos[4 x])

In[52]:= FactorList[x^5 + x^4 - 6 x^3 - x^2 - x + 6]
Out[52]= {{1, 1}, {-2 + x, 1}, {-1 + x, 1}, {3 + x, 1}, {1 + x + x^2, 1}}

In[53]:= FactorTerms[3 x^3 - 3 x^2 - 51 x - 45]
Out[53]= 3 (-15 - 17 x - x^2 + x^3)

In[54]:= FactorTermsList[3 x^3 - 3 x^2 - 51 x - 45]
Out[54]= {3, -15 - 17 x - x^2 + x^3}

In[55]:= FactorInteger[1235]
Out[55]= {{5, 1}, {13, 1}, {19, 1}}
```

Лістинг 9

Функції перетворення тригонометричних виразів

TrigReduce[f] – спрощує вираз f, який містить тригонометричні функції;

TrigExpand[f] – здійснює розширення виразів тригонометричних функцій;

TrigFactor[f] – розкладає на множники тригонометричні функції;

TrigToExp[f] – перетворює тригонометричний вираз в експоненційну форму;

ExpToTrig [f] – перетворює експоненційний вираз в тригонометричну форму.

В лістингу 9 показані приклади використання функцій перетворення тригонометричних виразів.

```
In[56]:= TrigReduce[(Sin[x] + Cos[x])^2]
Out[56]= 1 + Sin[2 x]

In[57]:= TrigReduce[2 Sin[1 / 2 (x + y)] Sin[1 / 2 (y - x)]]
Out[57]= Cos[x] - Cos[y]

In[58]:= TrigExpand[Sin[4 x]]
Out[58]= 4 Cos[x]^3 Sin[x] - 4 Cos[x] Sin[x]^3

In[62]:= TrigFactor[Cos[x] - Cos[y]]
Out[62]= -2 Sin[x/2 - y/2] Sin[x/2 + y/2]

In[63]:= TrigToExp[Cos[x] + Sin[x]]
Out[63]= (1/2 + i/2) e^(-i x) + (1/2 - i/2) e^(i x)

In[64]:= TrigToExp[1 / I * Sin[I * x]]
Out[64]= -e^(-x)/2 + e^x/2

In[65]:= ExpToTrig[E^x + E^(-x)]
Out[65]= 2 Cosh[x]
```

Лістинг 10

Хід роботи

1. Розрахувати факторіал числа 32.
2. Знайти суму дробів $\frac{2}{3}+\frac{7}{92}+\frac{3}{31}$ в форматі раціонального числа.
3. Знайти суму дробів $\frac{2}{3}+\frac{7}{92}+\frac{3}{31}$ в форматі дійсного числа.
4. Перевести дійсне число, отримане в п.3, в раціональне з точністю 10^{-4}.
5. Знайти модуль числа $(e^{5i}+32\cdot\cos(44+6i)-\sinh(33))$.
6. Записати аналітичний вираз функції $2\sin(1/2(x+y))\cos(1/2(x-y))$ і знайти значення функції при x=5, y=2.
7. Записати аналітичний вираз функції $1/2(1+1/i)e^{ix}+(1-1/i)e^{ix}$ і знайти значення функції при x=1..10 з кроком 1.
8. Приведіть подібні доданки у виразі $2x^5-6x^4+2x^3+11x^4-5+7x^2+6x^5-3$.
9. Розділіть $213x^6+213x^5+155x^4-58x^3-46x^2+12x+12$ на x^2+x+1.
10. Розкрийте дужки у виразі $(x+t-\tau)^2(t-\tau)$.
11. Розкладіть вираз $(x+y-1)^2+(x-\tau)^3+10\sin(y)\cdot(x-\tau)$ по степеням x.

Контрольні запитання

1. Перелічіть основні можливості системи Mathematica.
2. Опишіть основні типи даних системи Mathematica.
3. Вкажіть форму запису раціональних і дробових чисел і опишіть особливості їх використання.
4. Перелічіть основні функції, призначені для спрощення математичних виразів.
5. Наведіть приклади іменованих констант, які використовують у системі Mathematica.
6. Вкажіть команду для отримання оперативної довідку для конкретного об’єкта.
7. Опишіть особливості роботи в інтерактивному режимі.
8. Вкажіть призначення функції Collect[].
9. Назвіть функцію, яку доцільно використовувати для спрощення математичних виразів.
10. Вкажіть оператор для підстановки числових значень аргумента у аналітичну функцію.

Лабораторна робота № 2. Робота з векторами і матрицями

Мета: вивчення функцій системи Mathematica, що призначені для роботи з векторами і матрицями.

Короткі теоретичні відомості

Вектори і матриці в системі Mathematica є списками. Списком називається сукупність даних обмежена фігурними дужками. Вектор є одновимірним, матриця – двовимірним списком. Елементами векторів і матриць, можуть бути дійсні і уявні числа, функції, математичні вирази. В лістингу 1 наведені вектори і матриці з різних елементів.

```
In[1]:= f1 = {1, 2, 3, 4, 5}

Out[1]= {1, 2, 3, 4, 5}

In[2]:= f2 = {a, b, c, d, e}

Out[2]= {a, b, c, d, e}

In[3]:= f3 = {2 + 3 I, 1 - 2 I, 5, 3 + 7 I, 7}

Out[3]= {2 + 3 i, 1 - 2 i, 5, 3 + 7 i, 7}

In[4]:= f4 = {{1, 2, 3}, {4, 7, 0}, {-5, 1, 8}}

Out[4]= {{1, 2, 3}, {4, 7, 0}, {-5, 1, 8}}

In[5]:= f5 = {{Sin[x], E^-x, (x - 1) / (x + 1), Log[x], 5},
          {1 + 2 I, 3, 5, Tan[x + 1], -8}, {4, a, b, Cos[x], 2 - I}}
```

$$\text{Out[5]=}\ \left\{\left\{\text{Sin}[x],\ e^{-x},\ \frac{-1+x}{1+x},\ \text{Log}[x],\ 5\right\},\ \{1+2\,i,\ 3,\ 5,\ \text{Tan}[1+x],\ -8\},\ \{4,\ a,\ b,\ \text{Cos}[x],\ 2-i\}\right\}$$

Лістинг 1

Створення векторів і матриць здійснюється за допомогою наступних функцій.

Array [f, n]- створює вектор з n елементів f [1], f [2],…, f [n];

Array [f, n_1, n_2]- створює вектор з n_1 елементів, починаючи з f [n_2], при цьому n_2, може бути числом, функцією, виразом;

Array [f, { n_1, n_2}] - створює матрицю n_1 x n_2, з елементів функції f (n_1, n_2);

Array [f, n_1, n_2, h] - створює вектор з n_1 елементів, починаючи з f (n_2), з рівнем масиву h.

Приклади створення векторами за допомогою функції Array наведено у лістингу 2. Зверніть увагу на те, що ці функції дозволяють здійснювати підсумовування і множення елементів вектора за допомогою функцій Plus і Times.

```
In[21]:= Array[Tan, 5]
Out[21]= {Tan[1], Tan[2], Tan[3], Tan[4], Tan[5]}

In[22]:= Array[Sin, 5, 3]
Out[22]= {Sin[3], Sin[4], Sin[5], Sin[6], Sin[7]}

In[23]:= Array[Sin, 5, Sqrt[3]]
Out[23]= {Sin[√3], Sin[1 + √3], Sin[2 + √3], Sin[3 + √3], Sin[4 + √3]}

In[24]:= N[%]
Out[24]= {0.987027, 0.398189, -0.556742, -0.999807, -0.523654}

In[25]:= Array[Sin, 5, Sqrt[3], Plus]
Out[25]= Sin[√3] + Sin[1 + √3] + Sin[2 + √3] + Sin[3 + √3] + Sin[4 + √3]

In[26]:= N[%]
Out[26]= -0.694987

In[27]:= Array[Sin, 5, Sqrt[3], Times]
Out[27]= Sin[√3] Sin[1 + √3] Sin[2 + √3] Sin[3 + √3] Sin[4 + √3]

In[28]:= N[%]
Out[28]= -0.11456

In[30]:= Array[Tan, {3, 4}]
Out[30]= {{Tan[1, 1], Tan[1, 2], Tan[1, 3], Tan[1, 4]},
          {Tan[2, 1], Tan[2, 2], Tan[2, 3], Tan[2, 4]},
          {Tan[3, 1], Tan[3, 2], Tan[3, 3], Tan[3, 4]}}

In[31]:= Array[Exp, 5, 3, 2]
Out[31]= 2[e^3, e^4, e^5, e^6, e^7]

In[32]:= N[%]
Out[32]= 2.[20.0855, 54.5982, 148.413, 403.429, 1096.63]

In[33]:= Array[Exp, 5, a, h]
Out[33]= h[e^a, e^(1+a), e^(2+a), e^(3+a), e^(4+a)]
```

Лістинг 2

Виявлення структури вектора або матриці

Для виявлення структури вектора або матриці служать наступні функції:

VectorQ [V] - перевіряє, чи є V вектором, і видає True, якщо так, і False, якщо ні;

MatrixQ [M] - перевіряє, чи є м матрицею, і видає True, якщо так, і False, якщо ні;

Length [V] - повертає число елементів вектора V;

Length [M] - повертає число рядків матриці M;

MemberQ [V, n] - перевіряє, чи є у векторі V елемент n; якщо так, то видає True, якщо ні - False;

FreeQ [V, n] - перевіряє, чи має вектор V елемент n; якщо містить, то видає False, якщо ні - True;

FreeQ [M, n] - перевіряє, чи має матриця M елемент n; якщо містить, то видає False, якщо ні - True;

Dimensions [V] - повертає число елементів вектора V;

Dimensions [M] - повертає розмір матриці (кількість рядків і число стовпців);

Position [V, n] - повертає номери позицій елемента n вектора V;

Count [V, n] - повертає число елементів у векторі V, що мають значення n;

TensorRank [V] - видає ранг вектора V, якщо V є тензором;

TensorRank [M] - видає ранг матриці M, якщо м є тензором.

Перетворення і створення векторів і матриць

Система Mathematica має багаті можливості перетворення і створення нових векторів і матриць (і списків будь-якого рівня). Для цієї мети можуть використовуватись такі функції:

Drop [V, n] - видаляє перші n елементів вектора V;

Drop [V,-n]- видаляє останні n елементів вектора V;

Drop [V, {n}] – видаляє n-й елемент вектора V;

Drop [V, {m, n}]- видаляє елементи від m-го до n-го вектора V;

Last [f] - повертає останній елемент вектора (матриці) f;

Rest [V] - повертає вектор V з відкинутим першим елементом;

Take [V, n]- повертає вектор V з першими n елементами;

Take [V,-n]- повертає вектор V з останніми n елементами;

Take [V, {m, n}] - повертає вектор V з номерами елементів від m до n;

Append [V, a]- додає елемент a в кінець вектора V;

Prepend [V,-a]- додає елемент a в початок вектора V;

Insert [V, a, n] - вставляє елемент a в позицію n з відліком з початку вектора V;

Insert [V, a,-n]- вставляє елемент a в позицію n з відліком з кінця вектора V;

Delete [V, n]- видаляє елемент n з вектора V;

{Delete [f, n_1], Delete [f, n_2], Delete [f, n_3], ...} - видаляє з вектора (матриці) f елементи n_i та утворює новий вектор (матрицю).

Приклади використання функцій наведено у лістингу 3.

```
In[35]:= V = {1, 3, 4, 5, 7, 2, 4}
Out[35]= {1, 3, 4, 5, 7, 2, 4}

In[36]:= M = {{1, 3, 5}, {2, 4, 6}, {1, 2, 3}}
Out[36]= {{1, 3, 5}, {2, 4, 6}, {1, 2, 3}}

In[37]:= {Drop[V, 3], Drop[V, -4], Drop[V, {5}], Drop[V, {2, 5}]}
Out[37]= {{5, 7, 2, 4}, {1, 3, 4}, {1, 3, 4, 5, 2, 4}, {1, 2, 4}}

In[38]:= Delete[V, 6]
Out[38]= {1, 3, 4, 5, 7, 4}

In[39]:= {Delete[V, 1], Delete[V, -3], Delete[V, 7]}
Out[39]= {{3, 4, 5, 7, 2, 4}, {1, 3, 4, 5, 2, 4}, {1, 3, 4, 5, 7, 2}}

In[40]:= Delete[M, {2, 3}]
Out[40]= {{1, 3, 5}, {2, 4}, {1, 2, 3}}

In[41]:= Delete[M, {3, 1}]
Out[41]= {{1, 3, 5}, {2, 4, 6}, {2, 3}}

In[42]:= Last[V]
Out[42]= 4

In[43]:= Last[M]
Out[43]= {1, 2, 3}
In[44]:= Rest[V]
Out[44]= {3, 4, 5, 7, 2, 4}

In[45]:= {Take[V, 4], Take[V, -4], Take[V, {3, 6}]}
Out[45]= {{1, 3, 4, 5}, {5, 7, 2, 4}, {4, 5, 7, 2}}

In[46]:= Append[V, Sin[x]]
Out[46]= {1, 3, 4, 5, 7, 2, 4, Sin[x]}

In[47]:= Prepend[V, Sin[x]]
Out[47]= {Sin[x], 1, 3, 4, 5, 7, 2, 4}

In[48]:= Insert[V, a, 5]
Out[48]= {1, 3, 4, 5, a, 7, 2, 4}

In[49]:= Insert[V, {3, a, Sin[x]}, 5]
Out[49]= {1, 3, 4, 5, {3, a, Sin[x]}, 7, 2, 4}
```

Лістинг 3

Утворення нових векторів і матриць також можливе шляхом зміни розташування вектора або матриці, для чого використовуються наступні функції.

Flatten [M] - утворює вектор з будь-якого списку, наприклад, матриці;

Flatten [M, n]- утворює вектор у рядку n матриці M;

Sort [f] - сортує елементи вектора (матриці) f в природному порядку;

Reverse [f] - повертає вектор (матрицю) f в зворотному порядку розташування елементів;

RotateLeft [f] - повертає елементи вектора (матриці) f після одноразового повороту вліво;

RotateLeft [f, n] - повертає елементи вектора (матриці) f після п-кратного повороту вліво;

RotateRight [f] - повертає елементи вектора (матриці) f після одноразового повороту вправо;

RotateRight [f, n] - повертає елементи вектора (матриці) f після n-кратного повороту вправо;

Transpose [M] - зміна рядків і стовпців матриці.

Приклади використання функцій наведено у лістингу 4.

```
In[50]:= V = {1, 3, 6, 2, 4, 5}

Out[50]= {1, 3, 6, 2, 4, 5}

In[51]:= M = {{2, 1, 4}, {3, 8, 6}, {1, 2, 3}}

Out[51]= {{2, 1, 4}, {3, 8, 6}, {1, 2, 3}}

In[52]:= Flatten[M]

Out[52]= {2, 1, 4, 3, 8, 6, 1, 2, 3}

In[53]:= FlattenAt[M, 2]

Out[53]= {{2, 1, 4}, 3, 8, 6, {1, 2, 3}}

In[54]:= Sort[V]

Out[54]= {1, 2, 3, 4, 5, 6}

In[55]:= Sort[M]

Out[55]= {{1, 2, 3}, {2, 1, 4}, {3, 8, 6}}

In[56]:= Reverse[V]

Out[56]= {5, 4, 2, 6, 3, 1}

In[57]:= Reverse[M]

Out[57]= {{1, 2, 3}, {3, 8, 6}, {2, 1, 4}}

In[58]:= RotateLeft[V, 3]

Out[58]= {2, 4, 5, 1, 3, 6}
```

```
In[59]:= RotateRight[V, 2]

Out[59]= {4, 5, 1, 3, 6, 2}

In[60]:= RotateLeft[M, 1]

Out[60]= {{3, 8, 6}, {1, 2, 3}, {2, 1, 4}}

In[61]:= RotateRight[M, 1]

Out[61]= {{1, 2, 3}, {2, 1, 4}, {3, 8, 6}}

In[62]:= Transpose[M]

Out[62]= {{2, 3, 1}, {1, 8, 2}, {4, 6, 3}}
```

Лістинг 4

Представлення векторів і матриць в табличному вигляді можливо за допомогою функцій TableForm і MatrixForm, що мають вигляд:

TableForm[f]

Out[n] // MatrixForm

де f - ім'я вектора або матриці;

n - номер рядка, в якій знаходиться вектор або матриця;

% - застосовується у разі, якщо функція % / / MatrixForm подання вектора або матриці в табличному вигляді розташовується слідом за вектором (матрицею).

Вектор або матрицю можна також створити за допомогою функції List:

List [a, b, c,...] — створює вектор {a,b,c,...};

List [{a, b, c,..}, {d, e, f, ..}, {g, h, k, ..}] - створює матрицю {{a, b, c,..}, {d, e, f, ..}, {g, h, k, ..}}.

В лістингу 2 в табличній формі наведені вектор f3 та матриці f4, f5, з лістингу 1, утворені за допомогою функцій TableForm і MatrixForm.

```
In[63]:= TableForm[f3]

Out[63]//TableForm=
2 + 3 i
1 - 2 i
5
3 + 7 i
7

In[64]:= TableForm[f4]

Out[64]//TableForm=
1    2    3
4    7    0
-5   1    8

In[65]:= TableForm[f5]

Out[65]//TableForm=
Sin[x]     e^-x    (-1+x)/(1+x)   Log[x]       5
1 + 2 i    3       5              Tan[1 + x]   -8
4          a       b              Cos[x]       2 - i

In[67]:= Out[65] // MatrixForm

Out[67]//MatrixForm=
```

$$\begin{pmatrix} \mathrm{Sin}[x] & e^{-x} & \frac{-1+x}{1+x} & \mathrm{Log}[x] & 5 \\ 1+2i & 3 & 5 & \mathrm{Tan}[1+x] & -8 \\ 4 & a & b & \mathrm{Cos}[x] & 2-i \end{pmatrix}$$

Лістинг 5

Генерація векторів і матриць за допомогою функції Range

Функція Range застосовується для створення числових списків і має такі модифікації:

Range [n_{max}] - генерує вектор числових елементів виду {1, 2, ..., n_{max} };

Range [n_{min}, n_{max}] - генерує вектор з числових елементів виду {n_{min}, n_{max}};

Range [n_{min}, n_{max}, dn] - генерує вектор числових елементів від n_{min} до n_{max} з кроком dn.

Приклади реалізації функції наведено в лістингу 6.

```
In[69]:= Range[7]

Out[69]= {1, 2, 3, 4, 5, 6, 7}

In[70]:= Range[4, 10]

Out[70]= {4, 5, 6, 7, 8, 9, 10}

In[71]:= Range[3, 8, 0.5]

Out[71]= {3., 3.5, 4., 4.5, 5., 5.5, 6., 6.5, 7., 7.5, 8.}
```

Лістинг 6

Генерація векторів і матриць за допомогою функцій Table

Для створення векторів і матриць можна використовувати функцію Table, що має вигляд:

Table [f, { n_{max} }] - створює n_{max} копій виразу f;

Table [f,{1, n_{max}}] - створює список функцій f при в діапазоні від 1 до n_{max};

Table [f, {n, n_{min}, n_{max}}] — список функцій f в діапазоні від n_{min} до n_{max};

Table [f, {n, n_{min}, n_{max}, dn}]— створює список функцій f в діапазоні від n_{min} до n_{max} з кроком dn.

Приклад використання функції Table показаний у лістингу 7.

```
In[73]:= Table[Log[x], {5}]

Out[73]= {Log[x], Log[x], Log[x], Log[x], Log[x]}

In[74]:= Table[Log[x], {x, 5}]

Out[74]= {0, Log[2], Log[3], Log[4], Log[5]}

In[75]:= Table[Log[x], {x, 5, 10}]

Out[75]= {Log[5], Log[6], Log[7], Log[8], Log[9], Log[10]}

In[76]:= Table[Log[x], {x, 1, 3, 0.5}]

Out[76]= {0., 0.405465, 0.693147, 0.916291, 1.09861}

In[77]:= Table[i + j, {i, 2, 4}, {j, 2, 4}]

Out[77]= {{4, 5, 6}, {5, 6, 7}, {6, 7, 8}}
```

Лістинг 7

Виділення та виведення елементів вектора і матриці

В системі Mathematica реалізовано такі методи виділення елементів векторів і матриць:

- використання подвійних квадратних дужок;
- використання функції Part;
- використання функції Select.

Використання подвійних квадратних дужок

У цьому випадку вираз, що виокремлює елементи вектора або матриці, представляється у вигляді:

f [[n]] f [[n_1, n_2, ...]],

де f - ім'я вектора або матриці;

n - елемент що виділяється;

n_i - і-й елемент із сукупності елементів що виділяються.

Приклади виділення елементів показано у лістингу 8.

```
In[78]:= f1 = {2, 1, 4, 3, 5}

Out[78]= {2, 1, 4, 3, 5}

In[79]:= f2 = {{1, 2, 3}, {3, 5, 7}, {2, 4, 6}}

Out[79]= {{1, 2, 3}, {3, 5, 7}, {2, 4, 6}}

In[80]:= f3 = {a, 1, b, 2, 3, c}

Out[80]= {a, 1, b, 2, 3, c}

In[81]:= f4 = {2, a, Sin[x + y^2], b, 1}

Out[81]= {2, a, Sin[x + y^2], b, 1}

In[82]:= f1[[4]]

Out[82]= 3

In[83]:= f2[[2, 2]]

Out[83]= 5

In[84]:= f1[[{2, 3}]]

Out[84]= {1, 4}

In[85]:= {f2[[1, 2]], f2[[3, 1]]}

Out[85]= {2, 2}
```

Лістинг 8

Виділення елементів вектора і матриці за допомогою функції Part

Функція Part представляється в наступному вигляді:

{Part[f, n_1], Part[f, n_2],...},

де f - ім'я вектора;

n_i - i-й елемент вектора f.

У разі виділення елементів матриці функція Part має вигляд:

{Part[f, n_1, m_1], Part[f, n_2 , m_2],...},

де n_i - i-й елемент рядка матриці;

m_i - i-й елемент стовпчика матриці.

У разі виділення елементів зі складних елементів вектора або матриці функція Part представляється в наступному вигляді:

Part[f, n, m, 1],

де f - ім'я вектора або матриці;

n - номер елемента вектора f;

m - рівень виразу (m = 1 у разі вектора, m = 2 у разі матриці);

1 - номер елемента у векторі або матриці.

Приклади використання функції Part показано у лістингу 9.

```
In[87]:= f3 = {a, 1, b, 2, 3, c}
Out[87]= {a, 1, b, 2, 3, c}

In[88]:= f4 = {2, a, Sin[x + y^2], b, 1}
Out[88]= {2, a, Sin[x + y²], b, 1}

In[89]:= {Part[f3, 2], Part[f3, 5], Part[f3, 6]}
Out[89]= {1, 3, c}

In[90]:= {Part[f3, -3], Part[f3, -5], Part[f3, -6]}
Out[90]= {2, 1, a}

In[91]:= {Part[f4, 4], Part[f4, 3, 1, 1], Part[f4, 3, 1, 2]}
Out[91]= {b, x, y²}
```

Лістинг 9

Виведення елементів векторів і матриць здійснюється за допомогою функцій MatrixForm і TableForm.

Приклади використання цих форм виведення показана у лістингу 10. Використання опцій TableAlignments і TableSpacing, що дозволяють розташувати вектор і матрицю на екрані в бажаному вигляді показано у лістингу 11.

```
In[92]:= F = {{a, 1, 4}, {b, 2, 5}, {c, 3, b}}
Out[92]= {{a, 1, 4}, {b, 2, 5}, {c, 3, b}}

In[93]:= MatrixForm[F]
Out[93]//MatrixForm=
```

$$\begin{pmatrix} a & 1 & 4 \\ b & 2 & 5 \\ c & 3 & b \end{pmatrix}$$

```
In[94]:= TableForm[F]
Out[94]//TableForm=
a    1    4
b    2    5
c    3    b
```

Лістинг 10

```
In[96]:= s = {5, 73426813438765, 34}
Out[96]= {5, 73426813438765, 34}

In[97]:= TableForm[s, TableAlignments → Left]
Out[97]//TableForm=
         5
         73426813438765
         34

In[99]:= TableForm[s, TableAlignments → Center]

Out[99]//TableForm=
                5
         73426813438765
               34

In[101]:= s1 = {{1, 3, 4}, {5, 1, 1}, {3, 2, 1}}
Out[101]= {{1, 3, 4}, {5, 1, 1}, {3, 2, 1}}

In[102]:= TableForm[s1]
Out[102]//TableForm=
         1    3    4
         5    1    1
         3    2    1

In[103]:= TableForm[s1, TableSpacing → {1, 1}]
Out[103]//TableForm=
         1  3  4
         5  1  1
         3  2  1

In[105]:= TableForm[s1, TableSpacing → {5, 2}]
Out[105]//TableForm=
         1   3   4


         5   1   1


         3   2   1

In[106]:= TableForm[s1, TableSpacing → {2, 5}]
Out[106]//TableForm=
         1      3      4

         5      1      1

         3      2      1
```

Лістинг 11

Комбінування векторів і матриць

Комбінування векторів і матриць здійснюється за допомогою наступних функцій:

Union [F] - повертає новий вектор або матрицю, з якого вилучені повторювані елементи;

Union [f_1, f_2, ...] - об'єднує f_1, f_2 видаляючи повторювані елементи векторів і матриць;

Join [f_1, f_2...] - Об'єднує f_1, f_2 ... в єдиний ланцюг (конкатенація);

Complement [f_1, f_2, ...] - Повертає список f, елементи якого не містяться в f_1, f_2, ...;

Intersection [f_1, f_2, ...] - повертає впорядкований список елементів, загальних для всіх списків.

У лістингу 12 наведено приклади використання функцій.

```
In[107]:= V1 = {a, 3, 1, 7, 2, 4, 6}
Out[107]= {a, 3, 1, 7, 2, 4, 6}

In[108]:= V2 = {1, b, 3, 4, 5, 8}
Out[108]= {1, b, 3, 4, 5, 8}

In[109]:= V3 = {4, 5, c, 7, 8, 1}
Out[109]= {4, 5, c, 7, 8, 1}

In[110]:= M1 = {{1, 2, 4}, {3, 5, 8}, {7, 2, 3}}
Out[110]= {{1, 2, 4}, {3, 5, 8}, {7, 2, 3}}

In[111]:= M2 = {{a, b, c}, {1, 2, 3}, {4, 5, 6}}
Out[111]= {{a, b, c}, {1, 2, 3}, {4, 5, 6}}

In[112]:= Union[V1, V2, V3, M1, M2]
Out[112]= {1, 2, 3, 4, 5, 6, 7, 8, a, b, c, {1, 2, 3},
           {1, 2, 4}, {3, 5, 8}, {4, 5, 6}, {7, 2, 3}, {a, b, c}}

In[113]:= Union[{1, 2, 3, 1, 4, 5, 2, 4, 1, 7}]
Out[113]= {1, 2, 3, 4, 5, 7}

In[114]:= Join[V1, V2, V3, M1, M2]
Out[114]= {a, 3, 1, 7, 2, 4, 6, 1, b, 3, 4, 5, 8, 4, 5, c, 7, 8, 1,
           {1, 2, 4}, {3, 5, 8}, {7, 2, 3}, {a, b, c}, {1, 2, 3}, {4, 5, 6}}
```

```
In[115]:= Complement[V1, V2, V3]
Out[115]= {2, 6, a}

In[116]:= Intersection[V1, V2, V3]
Out[116]= {1, 4}
```

Лістинг 12

Математичні операції над векторами і матрицями

Найпростіші арифметичні операції над векторами і матрицями розглянемо на наступному прикладі.

Дано вектори V1, V2 і матриці Ml, M2:

V1 = {1, 3, 5, 2, 6, 4};

V2 = {2, 7, 5, 8, 1, 3};

M1 = {{1, 2, 3}, {6, 5, 4}, {1, 3, 5}};

M2 = {{3, 2, 1}, {4, 5, 6}, {5, 3, 1}}.

Необхідно:

- скласти, відняти, помножити і розділити вектор V1 і матрицю Ml на число 3;
- піднести в квадрат вектор V2 і матрицю M2;
- взяти квадратний корінь з вектора V1 і матриці Ml;
- обчислити e^{V1}, sin(V2), ln(M1), cosh(M1).

Функції, що реалізують математичні операції наведено в лістингу 13.

```
In[117]:= V1 = {1, 3, 5, 2, 6, 4}
Out[117]= {1, 3, 5, 2, 6, 4}

In[118]:= V2 = {2, 7, 5, 8, 1, 3}
Out[118]= {2, 7, 5, 8, 1, 3}

In[119]:= M1 = {{1, 2, 3}, {6, 5, 4}, {1, 3, 5}}
Out[119]= {{1, 2, 3}, {6, 5, 4}, {1, 3, 5}}

In[120]:= M2 = {{3, 2, 1}, {4, 5, 6}, {5, 3, 1}}
Out[120]= {{3, 2, 1}, {4, 5, 6}, {5, 3, 1}}

In[121]:= {V1 + 3, V1 - 3, V1 * 3, V1 / 3}
```

Out[121]= $\left\{\{4, 6, 8, 5, 9, 7\}, \{-2, 0, 2, -1, 3, 1\}, \{3, 9, 15, 6, 18, 12\}, \left\{\frac{1}{3}, 1, \frac{5}{3}, \frac{2}{3}, 2, \frac{4}{3}\right\}\right\}$

```
In[122]:= {M1 + 3, M1 - 3, M1 * 3, M1 / 3}
```

Out[122]= $\left\{\{\{4, 5, 6\}, \{9, 8, 7\}, \{4, 6, 8\}\}, \{\{-2, -1, 0\}, \{3, 2, 1\}, \{-2, 0, 2\}\}, \{\{3, 6, 9\}, \{18, 15, 12\}, \{3, 9, 15\}\}, \left\{\left\{\frac{1}{3}, \frac{2}{3}, 1\right\}, \left\{2, \frac{5}{3}, \frac{4}{3}\right\}, \left\{\frac{1}{3}, 1, \frac{5}{3}\right\}\right\}\right\}$

```
In[123]:= {V2^2, M2^2, Sqrt[V1], Sqrt[M1]}
```

Out[123]= $\left\{\{4, 49, 25, 64, 1, 9\}, \{\{9, 4, 1\}, \{16, 25, 36\}, \{25, 9, 1\}\}, \{1, \sqrt{3}, \sqrt{5}, \sqrt{2}, \sqrt{6}, 2\}, \{\{1, \sqrt{2}, \sqrt{3}\}, \{\sqrt{6}, \sqrt{5}, 2\}, \{1, \sqrt{3}, \sqrt{5}\}\}\right\}$

```
In[124]:= {E^V1, Sin[V2], Log[M1], Cosh[M2]}
```

Out[124]= $\{\{e, e^3, e^5, e^2, e^6, e^4\}, \{\mathrm{Sin}[2], \mathrm{Sin}[7], \mathrm{Sin}[5], \mathrm{Sin}[8], \mathrm{Sin}[1], \mathrm{Sin}[3]\}, \{\{0, \mathrm{Log}[2], \mathrm{Log}[3]\}, \{\mathrm{Log}[6], \mathrm{Log}[5], \mathrm{Log}[4]\}, \{0, \mathrm{Log}[3], \mathrm{Log}[5]\}\}, \{\{\mathrm{Cosh}[3], \mathrm{Cosh}[2], \mathrm{Cosh}[1]\}, \{\mathrm{Cosh}[4], \mathrm{Cosh}[5], \mathrm{Cosh}[6]\}, \{\mathrm{Cosh}[5], \mathrm{Cosh}[3], \mathrm{Cosh}[1]\}\}\}$

```
In[125]:= N[%]
Out[125]= {{2.71828, 20.0855, 148.413, 7.38906, 403.429, 54.5982},
  {0.909297, 0.656987, -0.958924, 0.989358, 0.841471, 0.14112},
  {{0., 0.693147, 1.09861}, {1.79176, 1.60944, 1.38629},
   {0., 1.09861, 1.60944}}, {{10.0677, 3.7622, 1.54308},
   {27.3082, 74.2099, 201.716}, {74.2099, 10.0677, 1.54308}}}
```

Лістинг 13

Інші функції, призначені для роботи з векторами і матрицями наведені нижче:

Det [M] - повертає детермінант (головний визначник) матриці;

IdentityMatrix [M] - повертає одиничну матрицю: матрицю з діагональними елементами, рівними 1, і рештою нулями;

Transpose [M] - повертає транспоновану матрицю, у якої рядки замінені стовпцями, а стовпці рядками;

Inverse [M] - повертає зворотну матрицю, тобто таку, яка будучи помноженою на вихідну дає одиничну матрицю;

Tr [M] - повертає слід матриці (суму діагональних елементів);

LinearSolve [M, b] - повертає вектор невідомих матричного рівняння M * x = b, де M - матриця коефіцієнтів системи рівнянь, x - вектор невідомих, b - вектор вільних членів;

Eigensystem[M] - повертає список власних значень і власних векторів квадратної матриці M;

Eigenvalues [M] - повертає список власних значень квадратної матриці M;

Eigenvectors[M] - повертає список власних векторів квадратної матриці M;

PseudoInverse[M] - шукає псевдоподібну матрицю для матриці M.

Всі перераховані вище матричні операції ілюструються у лістингу 14.

```
In[126]:= M1 = {{1, 2, 3}, {3, 5, 4}, {7, 2, 1}}

Out[126]= {{1, 2, 3}, {3, 5, 4}, {7, 2, 1}}

In[127]:= M2 = {{4, a, -2}, {1, 2, b}, {3, 5, c}}

Out[127]= {{4, a, -2}, {1, 2, b}, {3, 5, c}}

In[128]:= {Det[M1], Det[M2]}

Out[128]= {-40, 2 - 20 b + 3 a b + 8 c - a c}

In[129]:= IdentityMatrix[3]

Out[129]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

In[131]:= {Transpose[M1], Transpose[M2]}

Out[131]= {{{1, 3, 7}, {2, 5, 2}, {3, 4, 1}}, {{4, 1, 3}, {a, 2, 5}, {-2, b, c}}}
```

```
In[132]:= {Inverse[M1], Inverse[M2]}
```

$$\text{Out[132]=}\ \left\{\left\{\left\{\frac{3}{40}, -\frac{1}{10}, \frac{7}{40}\right\}, \left\{-\frac{5}{8}, \frac{1}{2}, -\frac{1}{8}\right\}, \left\{\frac{29}{40}, -\frac{3}{10}, \frac{1}{40}\right\}\right\}, \left\{\left\{\frac{-5b+2c}{2-20b+3ab+8c-ac}, \frac{-10-ac}{2-20b+3ab+8c-ac}, \frac{4+ab}{2-20b+3ab+8c-ac}\right\}, \left\{\frac{3b-c}{2-20b+3ab+8c-ac}, \frac{6+4c}{2-20b+3ab+8c-ac}, \frac{-2-4b}{2-20b+3ab+8c-ac}\right\}, \left\{-\frac{1}{2-20b+3ab+8c-ac}, \frac{-20+3a}{2-20b+3ab+8c-ac}, \frac{8-a}{2-20b+3ab+8c-ac}\right\}\right\}\right\}$$

```
In[133]:= {Tr[M1], Tr[M2]}

Out[133]= {7, 6 + c}

In[134]:= {LinearSolve[M1, {1, 3, 7}], LinearSolve[M2, {1, 2, 3}]}
```

$$\text{Out[134]=}\ \left\{\{1, 0, 0\}, \left\{\frac{-8-5b+3ab+2c-2ac}{2-20b+3ab+8c-ac}, \frac{6-9b+7c}{2-20b+3ab+8c-ac}, \frac{-17+3a}{2-20b+3ab+8c-ac}\right\}\right\}$$

```
In[135]:= PseudoInverse[M1]
```

$$\text{Out[135]=}\ \left\{\left\{\frac{3}{40}, -\frac{1}{10}, \frac{7}{40}\right\}, \left\{-\frac{5}{8}, \frac{1}{2}, -\frac{1}{8}\right\}, \left\{\frac{29}{40}, -\frac{3}{10}, \frac{1}{40}\right\}\right\}$$

```
In[140]:= Eigenvalues[M1]

Out[140]= {9.15..., -3.42..., 1.28...}

In[141]:= N[%]

Out[141]= {9.14592, -3.42345, 1.27752}

In[142]:= Eigenvectors[M1]
```

$$\text{Out[142]=}\ \left\{\left\{\frac{1}{71}\left(17+13\,(9.15\ldots)-(9.15\ldots)^2\right), \frac{1}{142}\left(-190-20\,(9.15\ldots)+7\,(9.15\ldots)^2\right), 1\right\}, \left\{\frac{1}{71}\left(17+13\,(-3.42\ldots)-(-3.42\ldots)^2\right), \frac{1}{142}\left(-190-20\,(-3.42\ldots)+7\,(-3.42\ldots)^2\right), 1\right\}, \left\{\frac{1}{71}\left(17+13\,(1.28\ldots)-(1.28\ldots)^2\right), \frac{1}{142}\left(-190-20\,(1.28\ldots)+7\,(1.28\ldots)^2\right), 1\right\}\right\}$$

```
In[143]:= N[%]
Out[143]= {{0.735903, 1.4973, 1.},
  {-0.552462, -0.278106, 1.}, {0.450362, -1.43751, 1.}}
```

Лістинг 14

Векторний добуток реалізується за допомогою функції Dot [V1, V2] або знаком множення у вигляді точки: V1. V2 або Ml. M2, див. лістинг 15.

```
In[144]:= V1 = {1, 3, 5, 2, 6, 4}
Out[144]= {1, 3, 5, 2, 6, 4}

In[145]:= V2 = {2, 7, 5, 8, 1, 3}
Out[145]= {2, 7, 5, 8, 1, 3}

In[146]:= M1 = {{1, 2, 3}, {6, 5, 4}, {1, 3, 5}}
Out[146]= {{1, 2, 3}, {6, 5, 4}, {1, 3, 5}}

In[147]:= M2 = {{3, 2, 1}, {4, 5, 6}, {5, 3, 1}}
Out[147]= {{3, 2, 1}, {4, 5, 6}, {5, 3, 1}}

In[148]:= V1*V2
Out[148]= {2, 21, 25, 16, 6, 12}

In[149]:= V1.V2
Out[149]= 82

In[150]:= M1.M2
Out[150]= {{26, 21, 16}, {58, 49, 40}, {40, 32, 24}}

In[151]:= Dot[V1, V2]
Out[151]= 82

In[152]:= Dot[M1, M2]
Out[152]= {{26, 21, 16}, {58, 49, 40}, {40, 32, 24}}
```

Лістинг 15

Хід роботи

1. Створити матрицю 4x4:

$$z = \begin{pmatrix} N+4 & N/2 & N\%2 & 0 \\ N/4 & N+3 & N\%3 & N\%4 \\ N\%2 & N\%3 & N+2 & N/6 \\ 0 & N\%4 & N/4 & N+1 \end{pmatrix},$$

де N – номер варіанту, % - знак операції залишок від ділення.

2. Представити матрицю z у табличній і матричній формі.
3. За допомогою функцій Insert [], Delete[] і Position[] замінити нульові елементи матриці на номер рядка, в якому вони знаходяться.
4. Знайти визначник матриці z.
5. Транспонувати матрицю z.
6. Знайти власні числа матриці z.
7. Створити з першого рядка і третього стовпчика матриці z вектор q.
8. Видалити з вектора q 3 і 8 елементи.
9. Об'єднати матрицю z і вектор q.
10. Здійснити сортування елементів утвореного списку.

Контрольні запитання

1. Вкажіть струкутуру системи Mathematica, яку використовують для створення векторів і матриць.
2. Перелічіть основні функції для створення векторів і матриць.
3. Назвіть функцію, яку використовують для представлення матриць у табличній формі.
4. Вкажіть функції, які використовують для виділення частин векторів і матриць.
5. Зазначте процедуру звертання до окремого елемента вектора або матриці у системі Mathematica.
6. Перелічіть основні математичні операції над векторами і матрицями.
7. Вкажіть функцію призначену для об'єднання векторів і матриць.
8. Вкажіть функцію призначену для сортування векторів і матриць.
9. Вкажіть функції призначені для видалення елементів векторів і матриць.
10. Вкажіть функцію призначену для вставки нових елементів у вектори і матриці.

Лабораторна робота № 3. Графічні функції системи Mathematica

Мета: вивчення графічних функцій системи Mathematica.

Короткі теоретичні відомості

Двовимірна графіка

Функція Plot

Функція Plot дозволяє будувати графіки, задані аналітично, в двовимірному просторі у прямокутній системі координат. На одному графіку може бути зображено декілька функцій. За замовчуванням на екран виводиться координатна сітка.

Формат запису функції Plot

Plot[f,{x,xmin,xmax}];

Plot[{f1,f2,...},{x,xmin,xmax}],

де f – функція, графік якої будується,

fi – i-а функція, графік якої будується, i=1,2,…

x – аргумент функції,

xmin, xmax – інтервал зміни аргументу x.

Опції функції Plot

Опції функції Plot задаються в наступному вигляді:

Ім’я опції -> значення опції.

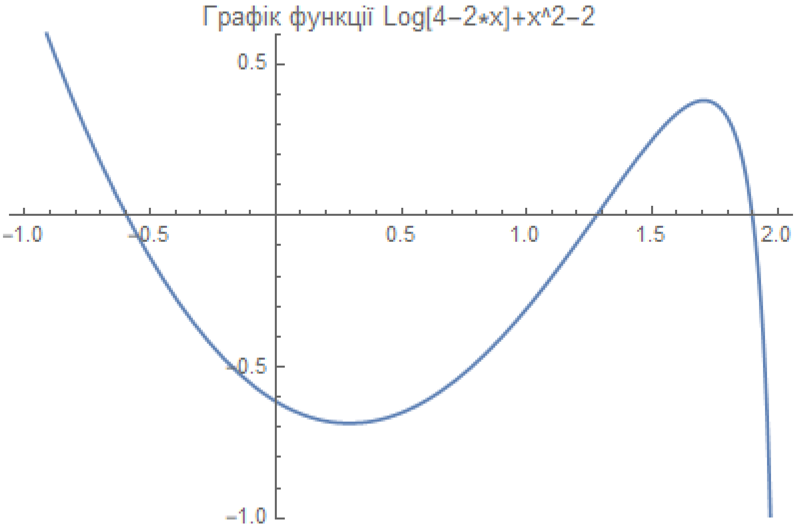
Основними опціями функції Plot є:

- встановлення масштабу по осі:
  PlotRange -> {ymin,ymax} – встановлює масштаб по осі y від ymin до ymax з автоматичним вибором кроку;
  PlotRange->{{ xmin,xmax}, {ymin,ymax}} - встановлює масштаб по осі y від ymin до ymax та по осі x від xmin до xmax з автоматичним вибором кроку;
- визначення імені осі:
  AxesLabel -> {“Tx”, “Ty”} – встановлює надписи Tx і Ty по осям x і y відповідно;
- визначення імені графіка:
  PlotLabel -> “T” – встановлює ім’я графіка;
- вибір стилю графіка:
  Axes -> None – графік будується без осей.

Приклади використання функції Plot.

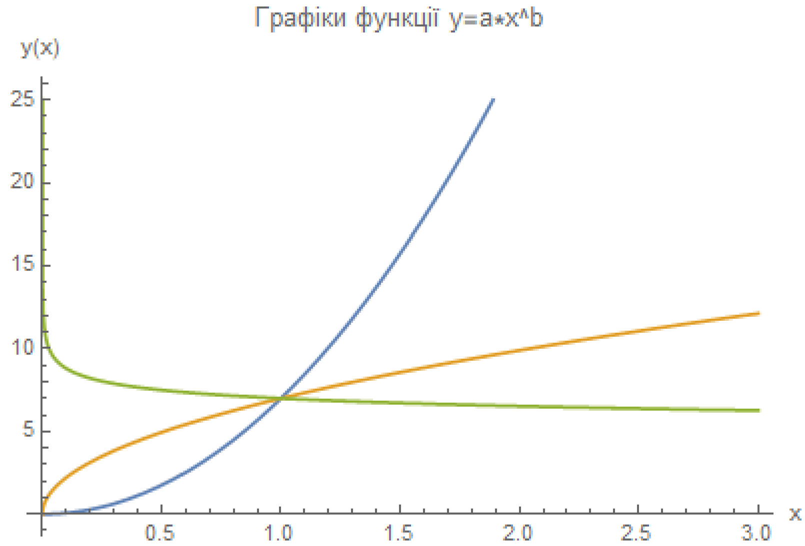
```
In[21]:= f = Log[4 - 2*x] + x^2 - 2;
         Plot[f, {x, -1, 2}, PlotRange → {-1, 0.6},
          PlotLabel → "Графік функції Log[4-2*x]+x^2-2"]
```

Out[22]=




```
In[23]:= Plot[{7*x^2, 7*x^0.5, 7*x^(-0.1)}, {x, 0, 3},
          AxesLabel → {"x", "y(x)"}, PlotLabel → "Графіки функції y=a*x^b"]
```

Out[23]=



Лістинг 1

Функція ListPlot

Використовується для побудови графіків, заданих у вигляді масиву точок.

Формат запису функції ListPlot

ListPlot[{y1,y2,…}];

ListPlot[{x1,y1},{x2,y2},…}],

де yi – i-е значення функції y(x),

xi - i-е значення аргументу функції y(x).

Призначення функції ListPlot

ListPlot[{y1,y2,…}] – виводить точкові значення функції y(x) з позначенням номеру точок по осі x.

ListPlot[{x1,y1},{x2,y2},…}] – виводить точкові значення функції y(x) з позначенням аргументу функції по осі x.

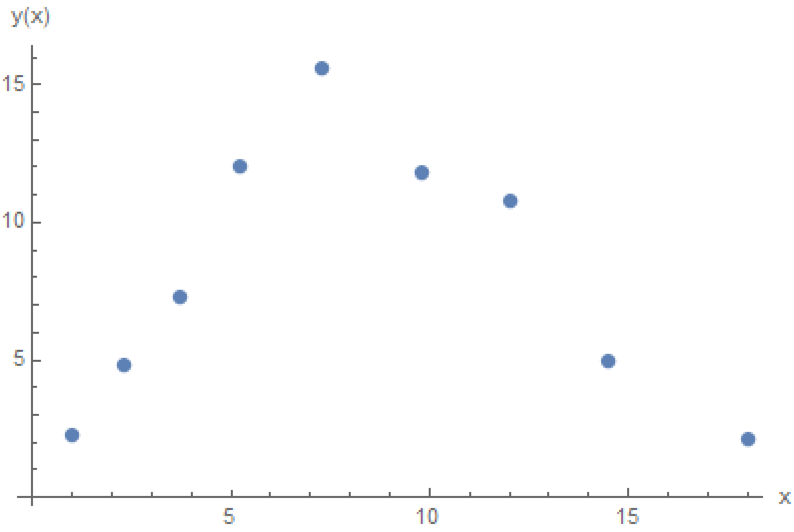
Функція ListPlot має дві опції:

- визначення імені осі:
  AxesLabel -> {“Tx”, “Ty”} – встановлює надписи Tx і Ty по осям x і y відповідно;
- визначення розміру точок:
  PlotStyle -> PointSize[d] – встановлює діаметр точки, рівний d.

Приклад використання функції ListPlot.

```
In[28]:= f = {{1, 2.3}, {2.3, 4.8}, {3.7, 7.3}, {5.2, 12}, {7.3, 15.6},
             {9.8, 11.8}, {12, 10.8}, {14.5, 5}, {18, 2.1}};
         ListPlot[f, AxesLabel → {"x", "y(x)"}, PlotStyle → PointSize[0.02]]
```

Out[29]=



Лістинг 2

Функція Show

Використовується для побудови точкових і аналітичних графіків в одній площині.

Формат запису функції Show

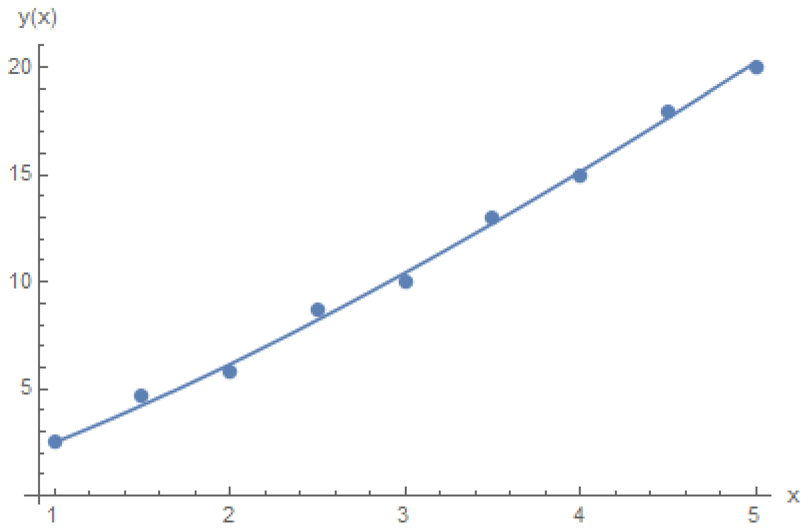
Show[r1,r2],

де r1, r2 – змінні, які використовуються для позначення графіків.

Приклад використання функції ListPlot.

```
In[30]:= f = {{1, 2.5}, {1.5, 4.7}, {2, 5.8}, {2.5, 8.7}, {3, 10}, {3.5, 13},
            {4, 15}, {4.5, 18}, {5, 20}};
         r1 := ListPlot[f, AxesLabel → {"x", "y(x)"},
            PlotStyle → PointSize[0.02]]
         r2 := Plot[2.5 x^1.3, {x, 1, 5}]
         Show[r1, r2]
```

Out[33]=



Лістинг 3

Вибір стилю графіка

Опція PlotStyle

Опція PlotStyle дозволяє обрати колір ліній і їх товщину.

Директиви опції PlotStyle

- колір лінії:

PlotStyle ->{GrayLevel[k1],GrayLevel[k2],…},

де k1,k2,… - коди кольору ліній у відтінках сірого відповідних функцій, обираються з діапазону 0..1;

PlotStyle ->{Hue[c1],Hue[c2],…},

де c1,c2,… - табличні коди кольору ліній відповідних функцій, обираються з діапазону 0..1;

PlotStyle ->{RGBColor[ч1,з1,c1],Hue[ч2,з2,c2],…},

де ч1,з1,c1… - яскравості червоної, зеленої і синьої компонент кольору, обираються з діапазону 0..1;

- товщина лінії:

PlotStyle ->Thickness[d] – встановлює товщину ліній графіка, як долю його повної ширини;

PlotStyle ->AbsoluteThickness[d] – встановлює товщину ліній графіка в пікселях;

- стиль штрихування:

PlotStyle ->Dashing[{d1,d2,… }] – встановлює довжину штриха ліній графіка, де di задається в частках від ширини лінії графіка;

PlotStyle->AbsoluteDashing[d1] - встановлює довжину штриха ліній графіка, де di задається в пікселях;

- точковий графік:

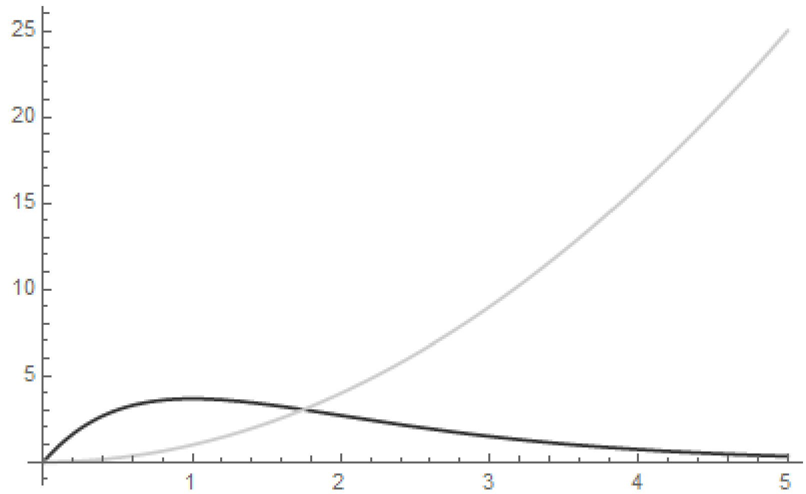
PlotStyle->PointSize[d] – графік у вигляді кіл діаметром d, які вимірюються в частках від загальної ширини графіка;

PlotStyle->AbsolutePointSize[d] – графік у вигляді кіл діаметром d, які вимірюються в пікселях.

Приклади використання опції PlotStyle.

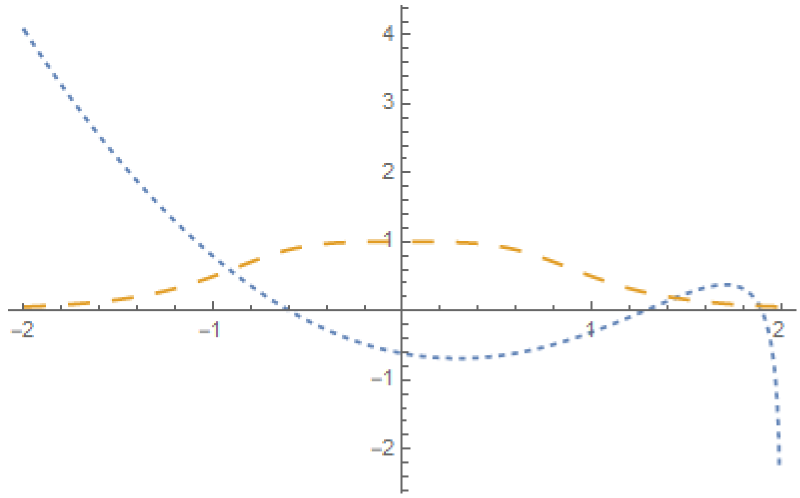
```
In[34]:= Plot[{10*x*Exp[-x], x^2}, {x, 0, 5},
  PlotStyle → {GrayLevel[0.2], GrayLevel[0.8]}]
```

Out[34]=



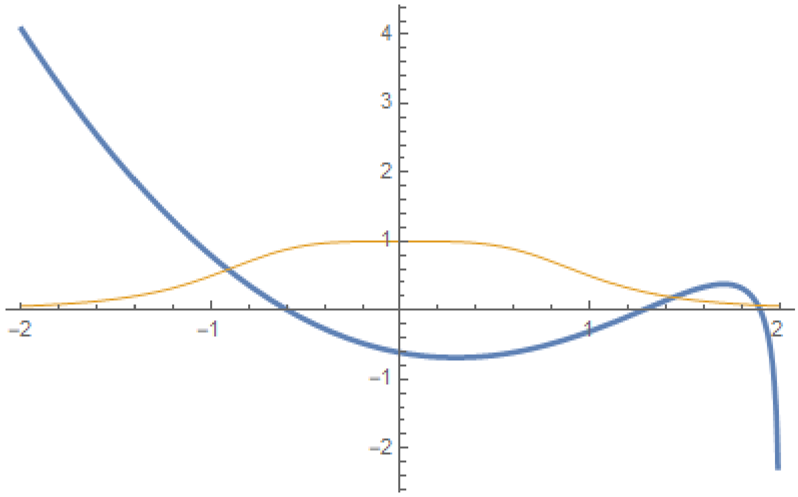
```
In[36]:= Plot[{Log[4 - 2*x] + x^2 - 2, 1 / (1 + x^4)}, {x, -2, 2},
  PlotStyle → {AbsoluteDashing[3], AbsoluteDashing[10]}]
```

Out[36]=



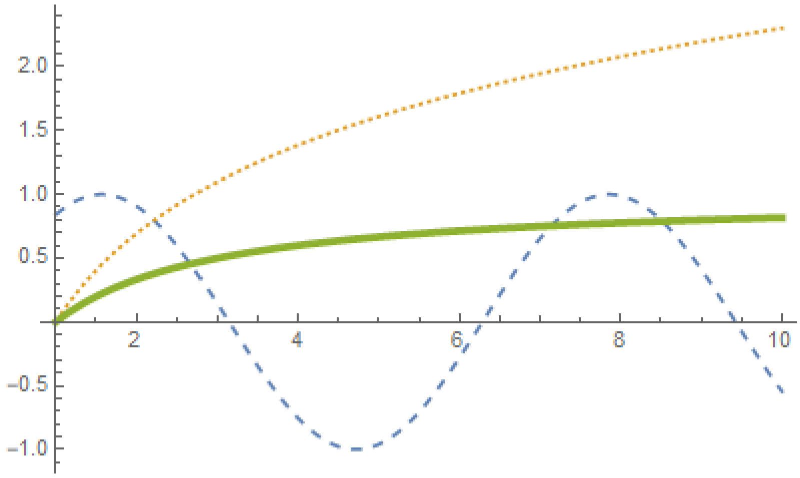
```
In[38]:= Plot[{Log[4 - 2*x] + x^2 - 2, 1 / (1 + x^4)}, {x, -2, 2},
   PlotStyle → {Thickness[0.007], Thickness[0.001]}]
```

Out[38]=




```
In[39]:= Plot[{Sin[x], Log[x], (x - 1) / (x + 1)}, {x, 1, 10},
   PlotStyle → {AbsoluteDashing[5], AbsoluteDashing[2], Thickness[0.01]}]
```

Out[39]=



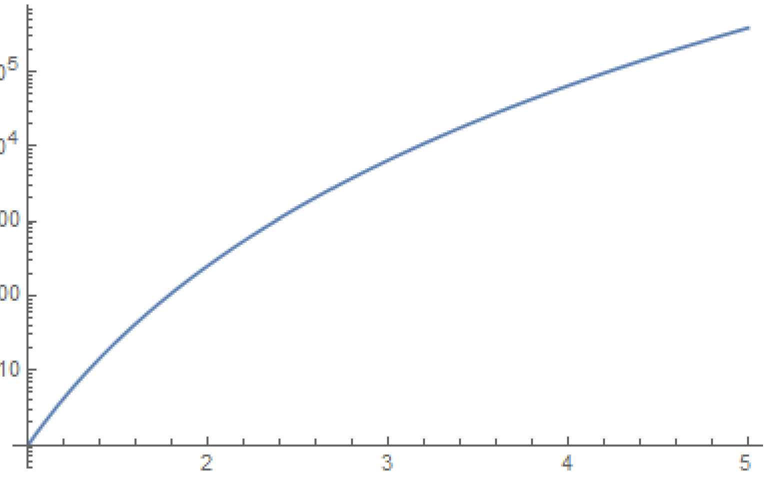
Лістинг 4

Графіки спеціальних типів

Функції побудови графіків в логарифмічному масштабі

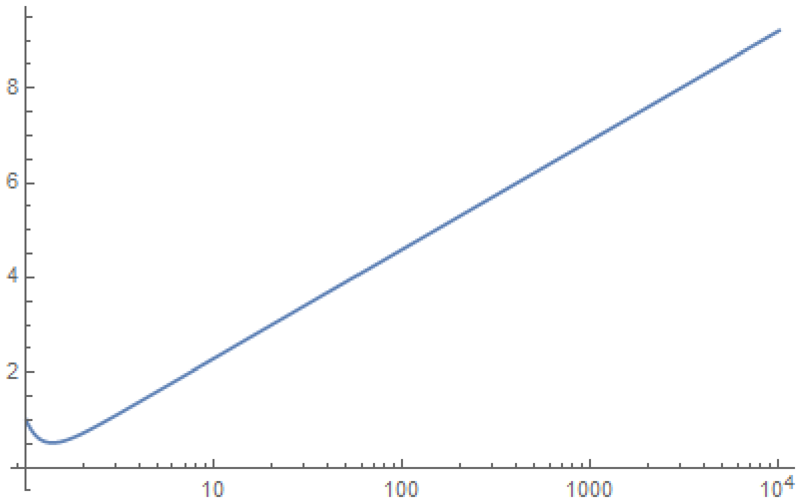
LogPlot[f,{x,xmin,xmax}] – створює лінійно-логарифмічний графік функції f з логарифмічною шкалою по осі y в діапазоні xmin..xmax.

In[41]:= LogPlot[x^8, {x, 1, 5}]

Out[41]=


LogLinearPlot[f,{x,xmin,xmax}] – створює логарифмічно-лінійний графік функції f з логарифмічною шкалою по осі x в діапазоні xmin..xmax.

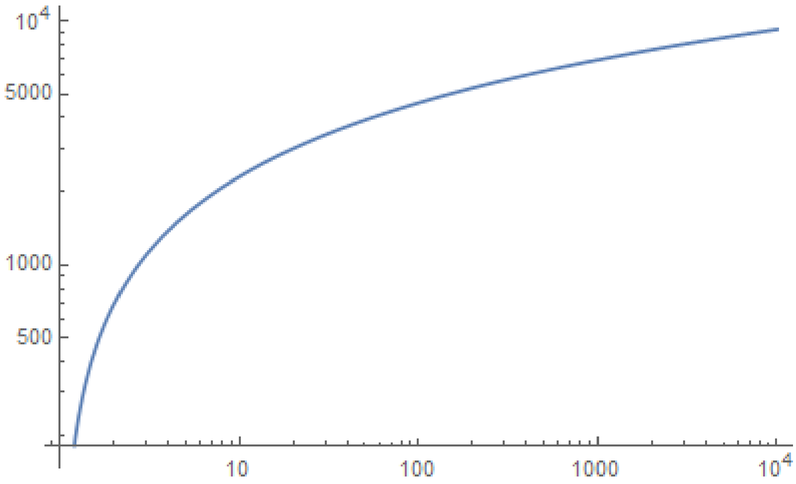
In[42]:= LogLinearPlot[x^-5 + Log[x], {x, 1, 10000}]

Out[42]=


LogLogPlot[f,{x,xmin,xmax}] – створює графік функції f з логарифмічною шкалою по двом осям в діапазоні xmin..xmax.

```
In[43]:= LogLogPlot[x^-5 + 1000*Log[x], {x, 1, 10000}]
```

Out[43]=



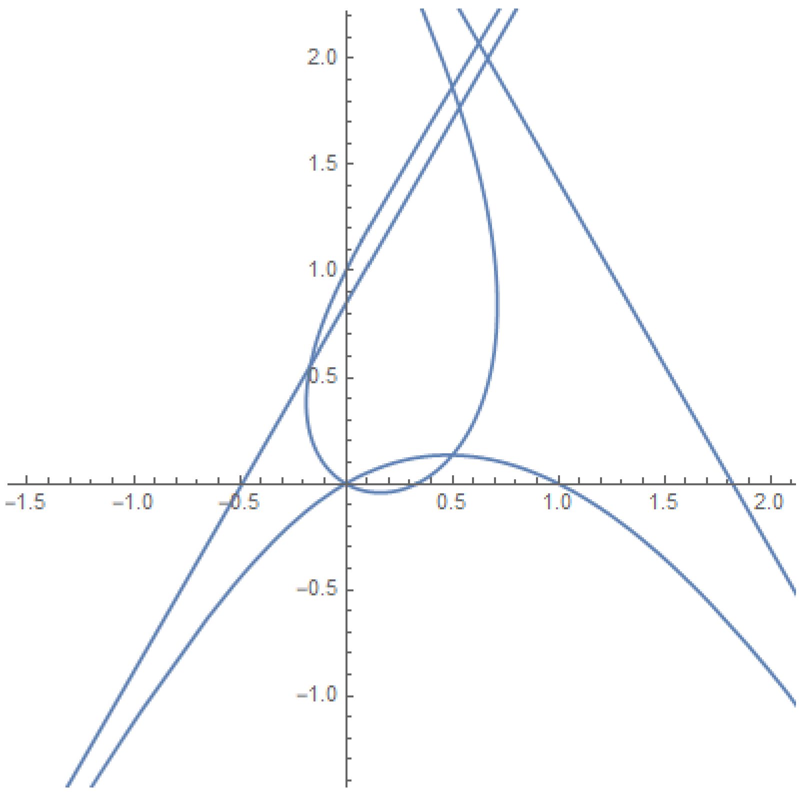
Функції
LogListPlot[{x1,y1},{x2,y2},…]
LogLinearListPlot[{x1,y1},{x2,y2},…]
LogLogListPlot[{x1,y1},{x2,y2},…]
аналогічні трьом попереднім і використовуються для побудови точкових графіків.

Функція побудови графіків в полярній системі координат

PolarPlot[f,{t,tmin,tmax}] – будує графік положення кінця вектора f при зміні кута t від tmin до tmax.

```
In[44]:= PolarPlot[(1 + 2*Sin[x]) / (1 + 2*Cos[x]), {x, 0, 2*Pi}]
```

Out[44]=

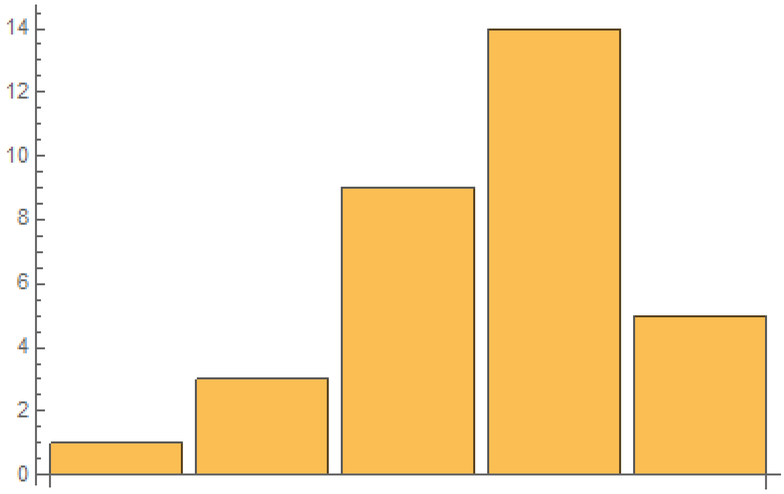


Функція побудови діаграм

BarChart[{c1,c2,…}] – будує діаграму у вигляді стовпців за даними списку.

In[46]:= `BarChart[{1, 3, 9, 14, 5}]`

Out[46]=

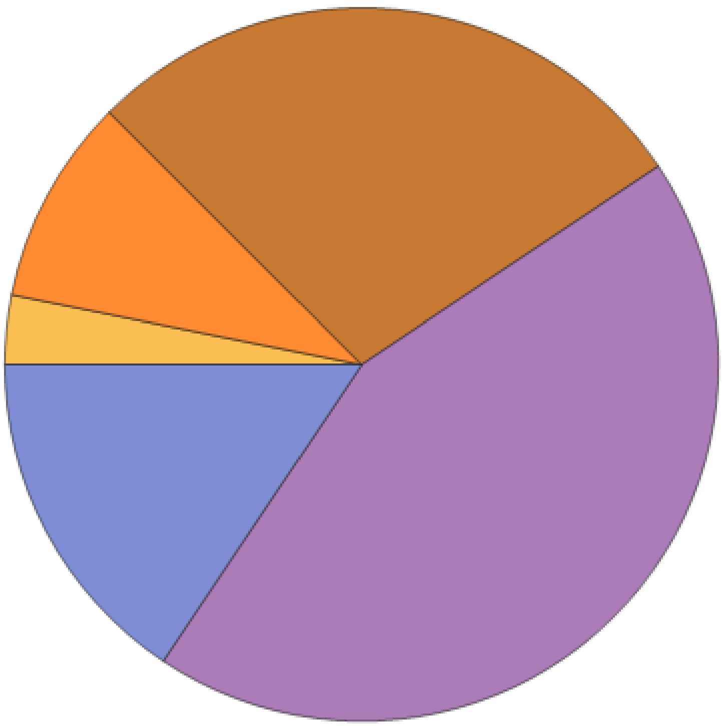


PieChart[{c1,c2,…}] – будує кругову діаграму за даними списку.

Разом з функцією PieChart використовується ряд опцій, опис яких доступний за командою Options[PieChart].

In[47]:= `PieChart[{1, 3, 9, 14, 5}]`

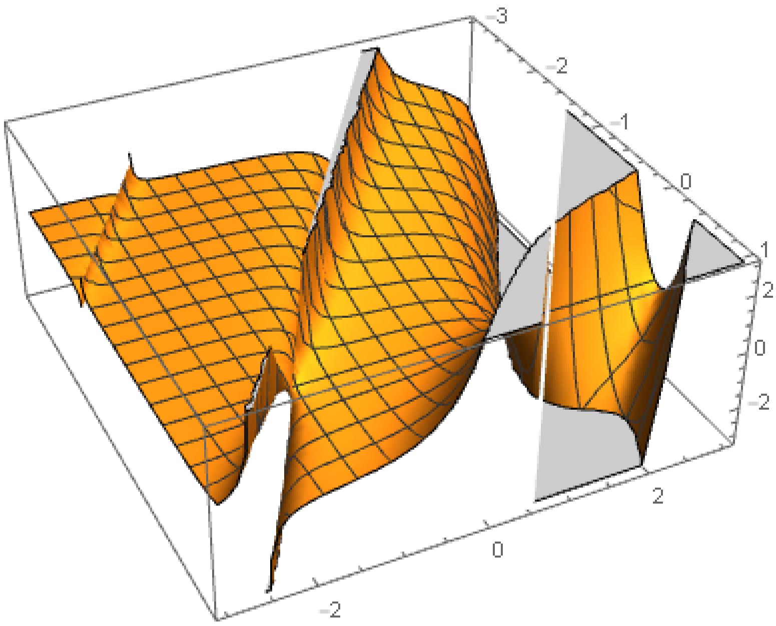
Out[47]=



Функції тривимірної графіки

Plot3D[f,{x,xmin,xmax},{y,ymin,ymax}] – будує графік функції f=f(x,y). Опції функції Plot3D доступні за командою Options[Plot3D].

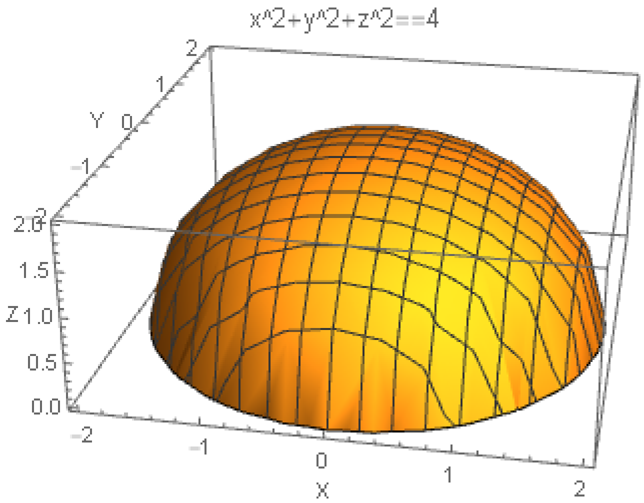
```
In[48]:= Plot3D[x*Exp[x+y]*Sin[x+y]/Cos[x+y], {x, -3, 1}, {y, -3, 3}]
```



При побудові графіків,що виражені через 3 аргументи потрібно виразити один аргумент через інші. Наприклад графік сфери: x^2+y^2+z^2=R^2, де R-радіус, перетвориться на $z = \sqrt{R^2 - x^2 - y^2}$

```
In[49]:= Plot3D[Sqrt[4 - x^2 - y^2], {x, -10, 10}, {y, -10, 10},
          AxesLabel → {"X", "Y", "Z"}, PlotLabel → "x^2+y^2+z^2==4"]
```

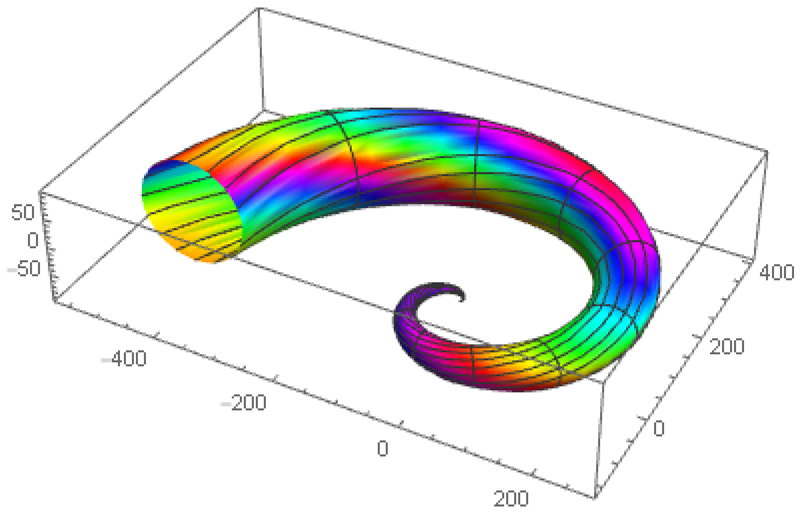
Out[49]=



ParametricPlot3D[{f1,f2,f3},{t1,t1min,t1max},{t2,t2min,t2max}] – будує тривимірний графік параметрично заданої функції z(t1,t2)=f(x(t1,t2),y(t1,t2)). Опції функції Plot3D доступні за командою Options[ParametricPlot3D].

```
In[63]:= ParametricPlot3D[{x^2*Cos[x]*(5+Cos[x+y]),
  x^2*Sin[x]*(5+Cos[x+y]), x^2*Sin[x+y]}, {x, 0, 3*Pi},
 {y, 0, 2*Pi},
 ColorFunction → Function[{x, y, z}, Hue[x^2+y^2+z^2]]]
```

Out[63]=



Хід роботи

1. Побудувати графіки неперервних функцій в одній площині координат:

$$y_1 = \frac{1}{N} \cdot x^2;$$

$$y_2 = \frac{N}{3} \cdot x \cdot e^{-x};$$

$$y_3 = N \cdot \ln x,$$

де N – номер варіанту.

Діапазон побудови графіків: x = 1..10.

Назва графіків: «Функції y1=…., y2=…., y3=….».

Підписи осей: «x», «y(x)».

Стилі ліній задані в табл.1.

Таблиця 1. Стилі ліній

Номер варіанту	Тип ліній (штрихова-ш, суцільна-с)	Колір	Довжина штриха, штрихової лінії, пікселів	Товщина лінії, пікселів
1	с-с-с	син.-черв.-зел.		1-3-6
2	ш-с-с	син.-фіол.-зел.	5	1-2-3
3	с-ш-с	син.-черв.-сір.	6	2-4-3
4	ш-ш-с	чор.-черв.-зел.	3-6	8-6-4
5	с-с-ш	син.-кор.-зел.	8	5-6-3
6	ш-с-ш	кор.-чор.-сір.	5-8	7-2-5
7	с-ш-ш	фіол.-черв.-чор.	3-7	4-3-6
8	ш-ш-ш	син.-черв.-зел.	1-3-6	2-9-3
9	с-с-с	син.-фіол.-зел.		4-8-6
10	ш-с-с	син.-черв.-сір.	8	1-8-3
11	с-ш-с	чор.-черв.-зел.	10	2-3-1
12	ш-ш-с	син.-кор.-зел.	3-7	2-6-5
13	с-с-ш	кор.-чор.-сір.	5	2-6-4

14	ш-с-ш	фіол.-черв.-чор.	1-4	2-3-6
15	с-ш-ш	зел.-блак.-фіол.	4-6	1-6-3

2. Побудувати графіки точкових функцій в одній площині координат:

$y_1 = N\sqrt{x}$;

$y_2 = \frac{3}{N} \cdot x \cdot \lg x$;

$y_3 = N \cdot \sin(x)$.

Діапазон побудови графіків: x = 1..10 з кроком Δx=1.

Підписи осей: «x», «y(x)».

Розмір точок графіків y_1 – ((N%4)*0.01, y_2 – ((N%5)*0.02, y_3 – ((N%3)*0.03, де операція % - залишок від ділення.

3. Побудувати графіки точкових і неперервних функцій п. 1,2 в одній площині координат.
4. Побудувати логарифмічний графік функції

$y = N \cdot \ln x$.

Діапазон зміни x = 1.10000.

Варіанти для яких виконується умова N%3=0 : по осі x – логарифмічний масштаб, y – логарифмічний масштаб.
N%3=1 : по осі x – логарифмічний масштаб, y – лінійний масштаб.
N%3=2 : по осі x – лінійний масштаб, y – логарифмічний масштаб.

5. Побудувати діаграму.

Для парних варіантів – кругову, непарних – у вигляді стовпців.
Дані: {N%3, (N+20)%7, (N+11)%2, (N+38)%22, (N+12)%5, (N+30)%8 }.

6. Побудувати графік функції в полярній системі координат

$y_2 = \frac{N}{3} e^{-Nx}$.

7. Побудувати тривимірний графік поверхні

$\frac{3y^2}{N} + \frac{2x^2}{N} - \frac{4z^2}{N} = 1$.

Контрольні запитання

1. Перелічіть основні функції для побудови графіків.
2. Назвіть основні опції функції Plot[].
3. Вкажіть функцію, яку необхідно використовувати для побудови графіків аналітичних і точкових функції на одному графічному полотні.
4. Опишіть можливі формати запису функції ListPlot[].
5. Перелічіть параметри оції PlotStyle[].
6. Перелічіть можливі параметри, які задають тип лінії графіка.
7. Перелічіть функції для побудови спеціальних графіків.
8. Назвіть функції, які використовують для побудови тривимірних графіків.
9. Назвіть функції, які використовують для побудови діаграм.

10. Перелічіть функції, які використовують для побудови логарифмічних графіків.

Лабораторна робота № 4. Функції розв'язання алгебраїчних рівнянь і систем рівнянь в програмі Mathematica

Мета: отримати навички з розв'язання алгебраїчних рівнянь і систем рівнянь в програмі Mathematica.

Короткі теоретичні відомості

Аналітичні методи розв'язку алгебраїчних і трансцендентних рівнянь

Функція Solve

Для розв'язку рівнянь у аналітичному вигляді застосовується функція Solve, формат запису якої є наступним:

Solve[f,x],

де f – рівняння, яке записується у довільному вигляді,
x – ім'я змінної.

Для запису рівняння використовується символ рівності «==», наприклад $ax^2 + bx + c == 0$. Необхідно мати на увазі, що функція Solve не завжди формує рішення у найкомпактнішому виді, тому після використання цієї функції іноді потрібно використовувати функції Simplify, Expand або FullSimplify. Приклади використання функції Solve наведено у лістингу 1.

```
In[22]:= f1 := x^5 + b*x^4 - a^4*x - a^4*b == 0
         Solve[f1, x]

Out[23]= {{x → -a}, {x → -i a}, {x → i a}, {x → a}, {x → -b}}

In[24]:= f2 := Sin[a*x] + Cos[a*x] == 0
         Solve[f2, x]
```

$$\text{Out[25]= }\left\{\left\{x \to \text{ConditionalExpression}\left[\frac{-\frac{\pi}{4} + 2\pi c_1}{a},\ c_1 \in \mathbb{Z}\right]\right\},\right.$$
$$\left.\left\{x \to \text{ConditionalExpression}\left[\frac{\frac{3\pi}{4} + 2\pi c_1}{a},\ c_1 \in \mathbb{Z}\right]\right\}\right\}$$

Лістинг 1

Функція Roots

Ця функція призначена для визначення коренів полінома, вона має вигляд:

Roots[f, x],

де f – поліном, корені якого необхідно знайти (може бути представлений у вигляді рівняння),
x – аргумент полінома f.

Результатом застосування функції Roots[f, x] є дійсні і комплексні корені рівняння $f(x) = 0$. При цьому розв'язок може бути отриманий в аналітичному і чисельному вигляді. Розв'язок в аналітичному вигляді в загальному випадку може бути отриманий для полінома не вище четвертого ступеня. Розв'язок

$f(x)=0$ не існує, якщо $f(x)$ – поліном п'ятого і вище ступеня. Однак якщо поліном може бути розкладений на множники, то функція Roots[f, x] знайде всі корені відповідного рівняння.

Розв'язки рівнянь у зазначених випадках показано на прикладах в лістингу 2 для рівнянь: $ax^2+bx+c=0$, $ax^4+bx^2+c=0$, $ax^3+b=0$, $ax^5+bx^4+cx+d=0$, $(a+x)(b+x)(x+ac)(x+1)(x-1)$. З лістингу 2 видно, що в перших трьох прикладах корені знайдено в аналітичному вигляді. У четвертому прикладі наведено поліном п'ятого ступеня, тому його корені не знайдені. В останньому прикладі наведено поліном п'ятого ступеня, при цьому рішення отримано в аналітичному вигляді. Це пояснюється тим, що многочлен $f(x)$ може бути розкладений на множники. Після цього за допомогою функції Expand розкрито дужки многочлена, і знайдено всі його корені за допомогою функції Roots[Out[32], x]. Тут Out[32] – номер рядка, в якій знаходиться поліном в розкритому вигляді.

In[26]:= Roots[a x^2 + b x + c == 0, x]

Out[26]= $x == \frac{-b-\sqrt{b^2-4ac}}{2a} \;||\; x == \frac{-b+\sqrt{b^2-4ac}}{2a}$

In[27]:= Roots[a x^4 + b x^2 + c == 0, x]

Out[27]= $x == \frac{\sqrt{-\frac{b}{a}-\frac{\sqrt{b^2-4ac}}{a}}}{\sqrt{2}} \;||\; x == -\frac{\sqrt{-\frac{b}{a}-\frac{\sqrt{b^2-4ac}}{a}}}{\sqrt{2}} \;||$

$x == \frac{\sqrt{-\frac{b}{a}+\frac{\sqrt{b^2-4ac}}{a}}}{\sqrt{2}} \;||\; x == -\frac{\sqrt{-\frac{b}{a}+\frac{\sqrt{b^2-4ac}}{a}}}{\sqrt{2}}$

In[29]:= Roots[a x^3 + b == 0, x]

Out[29]= $x == -\frac{(-1)^{2/3} b^{1/3}}{a^{1/3}} \;||\; x == \frac{(-1)^{1/3} b^{1/3}}{a^{1/3}} \;||\; x == -\frac{b^{1/3}}{a^{1/3}}$

In[30]:= Roots[a x^5 + b x^4 + c x + d == 0, x]

Out[30]= $x == \mathrm{Root}[d + c \#1 + b \#1^4 + a \#1^5 \&, 1] \;||$
$x == \mathrm{Root}[d + c \#1 + b \#1^4 + a \#1^5 \&, 2] \;||$
$x == \mathrm{Root}[d + c \#1 + b \#1^4 + a \#1^5 \&, 3] \;||$
$x == \mathrm{Root}[d + c \#1 + b \#1^4 + a \#1^5 \&, 4] \;||$
$x == \mathrm{Root}[d + c \#1 + b \#1^4 + a \#1^5 \&, 5]$

In[31]:= (a + x) (b + x) (x + ac) (x + 1) (x - 1)

Out[31]= (-1 + x) (1 + x) (a + x) (ac + x) (b + x)

In[32]:= Expand[Out[31]]

Out[32]= $-a\,ac\,b - a\,ac\,x - a\,b\,x - ac\,b\,x - a\,x^2 - ac\,x^2 - b\,x^2 +$
$a\,ac\,b\,x^2 - x^3 + a\,ac\,x^3 + a\,b\,x^3 + ac\,b\,x^3 + a\,x^4 + ac\,x^4 + b\,x^4 + x^5$

In[33]:= Roots[Out[32] == 0, x]

Out[33]= x == -b || x == -ac || x == -a || x == -1 || x == 1

Лістинг 2

Якщо коефіцієнти полінома задані у вигляді чисел, то функція Roots[f, x] видає розв'язок у вигляді точного або наближеного значення коренів. Точне значення коренів представляється числами в раціональній формі, наближене – у формі дійсних чисел. В лістингу 3 наведено приклади вирішення таких рівнянь: $x^2+13/28x-3/14=0$, $2x^7+3x^6-12x^2+7x+5=0$, $x^{10}-1=0$. З лістингу 3 видно, що функція визначила точне значення коренів першого рівняння, не змогла знайти в явному вигляді корені другого рівняння і знайшла корені третього рівняння, але у незручному для користувача вигляді: відсутні корені в комплексній формі. Для отримання розв'язку другого і третього рівнянь довелося використовувати команду N[%].

```
In[34]:= Roots[x^2 + 13 / 28 x - 3 / 14 == 0, x]

Out[34]= x == 2/7 || x == -3/4

In[35]:= Roots[2 x^7 + 3 x^6 - 12 x^2 + 7 x + 5 == 0, x]

Out[35]= x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 1, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 2, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 3, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 4, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 5, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 6, 0] ||
  x == Root[2 #1^7 + 3 #1^6 - 12 #1^2 + 7 #1 + 5 &, 7, 0]

In[36]:= N[%]

Out[36]= x == -0.417339 || x == -1.61472 - 0.729406 i ||
  x == -1.61472 + 0.729406 i ||
  x == 0.11472 - 1.37166 i || x == 0.11472 + 1.37166 i ||
  x == 0.958665 - 0.296826 i || x == 0.958665 + 0.296826 i

In[37]:= Roots[x^10 - 1 == 0, x]

Out[37]= x == 1 || x == (-1)^(1/5) || x == (-1)^(2/5) ||
  x == (-1)^(3/5) || x == (-1)^(4/5) || x == -1 || x == -(-1)^(1/5) ||
  x == -(-1)^(2/5) || x == -(-1)^(3/5) || x == -(-1)^(4/5)

In[38]:= N[%]

Out[38]= x == 1. || x == 0.809017 + 0.587785 i ||
  x == 0.309017 + 0.951057 i || x == -0.309017 + 0.951057 i ||
  x == -0.809017 + 0.587785 i || x == -1. ||
  x == -0.809017 - 0.587785 i || x == -0.309017 - 0.951057 i ||
  x == 0.309017 - 0.951057 i || x == 0.809017 - 0.587785 i
```

Лістинг 3

Застосування функції Roots[f, x] до трансцендентних рівнянь дає помилковий результат, тому для цього типу рівнянь її застосування нераціональне.

Чисельні методи розв'язку алгебраїчних і трансцендентних рівнянь

Існує велика кількість чисельних методів розв'язання алгебраїчних і трансцендентних рівнянь. Алгоритм будь-якого з цих методів є сукупністю умов вибору початкового наближення, розрахункових співвідношень і ознаки закінчення обчислювального процесу.

Система Mathematica має багато вбудованих функцій розв'язання алгебраїчних і трансцендентних рівнянь в чисельному вигляді. Основними з них є: NSolve, NRoots, FindRoot. Розглянемо детально ці функції і наведемо приклади.

Функція NSolve

Функція NSolve має вигляд:

NSolve[f, x],

де f – вирішуване рівняння, x – шукане невідоме.

Результатом виконання цієї функції є корені рівняння $f(x)=0$. Коренями можуть бути дійсні та комплексні числа. Функція NSolve[f, x]може вирішувати всі рівняння, які вирішує функція Solve. Її відмінність лише у формі подання відповідей. Правила використання функції NSolve показані в лістингу 4 при вирішенні таких рівняннях: $x^3+9/4x^2-3/4x+5/16=0$, $2^x-4x+1=0$, $e^{-2x}+3/x-1=0$, $ax^3-1=0$, $2x^5+3.2x^3-7.3x-14=0$.

Функція NRoots

Функція NRoots має вигляд:

NRoots[f, x],

де f – вирішуване рівняння, x – шукане невідоме.

Результатом використання цієї функції є корені полінома $f(x)=0$. Правила використання функції NRoots показані в лістингу 5 при вирішенні таких рівняннях: $x^2+13/28x-3/14=0$, $2x^7+3x^6-12x^2+5=0$, $x^{10}-1=0$, $2^x-4x+1=0$. З лістингу 5 видно, що дійсні та комплексні корені знайдені в чисельному вигляді. Спроба вирішити трансцендентне рівняння $2^x-4x+1=0$ до успіху не привела – розв'язок не отримано.

Функція FindRoot

Функція FindRoot має вигляд:

FindRoot[f, {x, x_0}],

де f – рівняння, x – шукане невідоме (корінь (корені)) рівняння, x_0 – початкове наближення.

Функція FindRoot[f, {x, x_0}] знаходить корінь рівняння $f(x)=0$ з області значень x, близьких до x_0. Рекомендується наступна методика визначення коренів алгебраїчних і трансцендентних рівнянь за допомогою функції FindRoot[f, {x, x_0}]:

1. Визначення області ізоляції шуканого кореня і вибір значення наближення x_0.
2. Введення рівняння $f(x)=0$ з присвоєнням йому унікального імені.

3. Введення функції FindRoot[f, {x, x_0}] з обраним значенням x_0.

4. Отримання розв'язку натисканням комбінації клавіш <Shift>+<Enter>.

В якості приклада розглянута послідовність дій при знаходженні коренів рівняння $3^x - 9x + 1 = 0$. За графіком функції, зображеного рис. 1, обрано перші наближення для коренів функції.

```
In[39]:= Solve[x^3 - 2.25 x^2 - 0.75 x + 0.3125 == 0, x]

Out[39]= {{x -> -0.5}, {x -> 0.25}, {x -> 2.5}}

In[40]:= Solve[x^3 - 9/4 x^2 - 3/4 x + 5/16 == 0, x]
```

Out[40]= $\left\{\left\{x \to -\frac{1}{2}\right\}, \left\{x \to \frac{1}{4}\right\}, \left\{x \to \frac{5}{2}\right\}\right\}$

```
In[41]:= NSolve[x^3 - 9/4 x^2 - 3/4 x + 5/16 == 0, x]

Out[41]= {{x -> -0.5}, {x -> 0.25}, {x -> 2.5}}

In[43]:= Solve[2^x - 4 x + 1 == 0, x]
```

Out[43]= $\left\{\left\{x \to \frac{\text{Log}[2] - 4\,\text{ProductLog}\left[-\frac{\text{Log}[2]}{2\,2^{3/4}}\right]}{4\,\text{Log}[2]}\right\}, \left\{x \to \frac{\text{Log}[2] - 4\,\text{ProductLog}\left[-1, -\frac{\text{Log}[2]}{2\,2^{3/4}}\right]}{4\,\text{Log}[2]}\right\}\right\}$

```
In[45]:= NSolve[2^x - 4 x + 1 == 0, x]

Out[45]= {{x -> 0.639428}, {x -> 3.84666}}
```

Лістинг 4 (Частина 1 з 2)

```
In[48]:= Solve[E^(-2 x) + 3/x - 1, x]
```

Solve: $-1+e^{-2x}+\frac{3}{x}$ is not a quantified system of equations and inequalities.

Out[48]= $\text{Solve}\left[-1+e^{-2x}+\frac{3}{x},\ x\right]$

```
In[49]:= NSolve[E^(-2 x) + 3/x - 1, x]
```

NSolve: This system cannot be solved with the methods available to NSolve.

Out[49]= $\text{NSolve}\left[-1+e^{-2x}+\frac{3}{x},\ x\right]$

```
In[50]:= Solve[a x^3 - 1 == 0, x]
```

Out[50]= $\left\{\left\{x\to\frac{1}{a^{1/3}}\right\},\ \left\{x\to-\frac{(-1)^{1/3}}{a^{1/3}}\right\},\ \left\{x\to\frac{(-1)^{2/3}}{a^{1/3}}\right\}\right\}$

```
In[51]:= NSolve[a x^3 - 1 == 0, x]
```

Out[51]= $\left\{\left\{x\to\frac{1}{a^{1/3}}\right\},\ \left\{x\to-\frac{0.5+0.866025\,i}{a^{1/3}}\right\},\ \left\{x\to-\frac{0.5-0.866025\,i}{a^{1/3}}\right\}\right\}$

```
In[52]:= Solve[2 x^5 - 3.2 x^3 + 7.3 x - 14 == 0, x]

Out[52]= {{x → -1.43009 - 0.904276 i},
  {x → -1.43009 + 0.904276 i}, {x → 0.699575 - 1.08818 i},
  {x → 0.699575 + 1.08818 i}, {x → 1.46103}}

In[53]:= NSolve[2 x^5 - 3.2 x^3 + 7.3 x - 14 == 0, x]

Out[53]= {{x → -1.43009 - 0.904276 i},
  {x → -1.43009 + 0.904276 i}, {x → 0.699575 - 1.08818 i},
  {x → 0.699575 + 1.08818 i}, {x → 1.46103}}
```

Лістинг 4 (Частина 2 з 2)

```
In[54]:= NRoots[x^2 + 13 / 28 x - 3 / 14 == 0, x]

Out[54]= x == -0.75 || x == 0.285714

In[55]:= NRoots[2 x^7 + 3 x^6 - 12 x^2 + 7 x + 5 == 0, x]

Out[55]= x == -1.61472 - 0.729406 i ||
          x == -1.61472 + 0.729406 i || x == -0.417339 ||
          x == 0.11472 - 1.37166 i || x == 0.11472 + 1.37166 i ||
          x == 0.958665 - 0.296826 i || x == 0.958665 + 0.296826 i

In[56]:= NRoots[x^10 - 1 == 0, x]

Out[56]= x == -1. || x == -0.809017 - 0.587785 i ||
          x == -0.809017 + 0.587785 i || x == -0.309017 - 0.951057 i ||
          x == -0.309017 + 0.951057 i || x == 0.309017 - 0.951057 i ||
          x == 0.309017 + 0.951057 i || x == 0.809017 - 0.587785 i ||
          x == 0.809017 + 0.587785 i || x == 1.

In[57]:= NRoots[2^x - 4 x + 1 == 0, x]

    NRoots: 1 + 2^x - 4 x == 0 is expected to be a polynomial equation in the variable x with
        numeric coefficients.

Out[57]= NRoots[1 + 2^x - 4 x == 0, x]
```

Лістинг 5

```
In[58]:= Plot[3^x - 9 x + 1, {x, 0, 4}]

Out[58]=
```

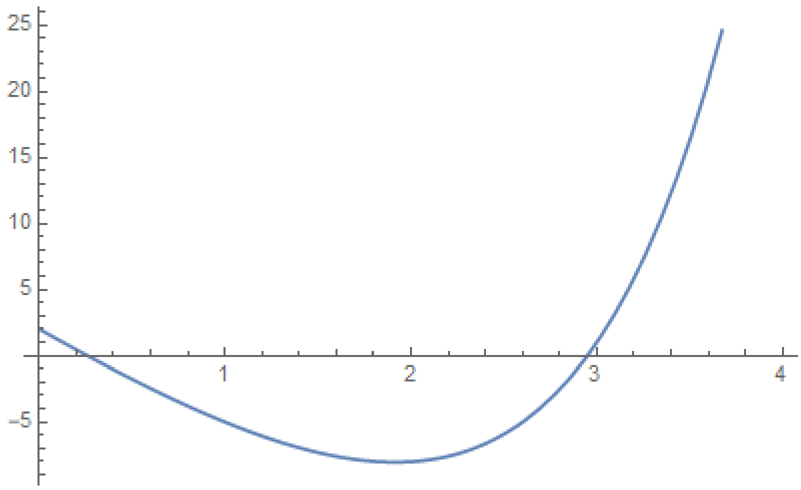


Рис. 1. Графік функції $f(x) = 3^x - 9x + 1$

З рис. 1 видно, що рівняння має два кореня, областями ізоляції яких можуть бути інтервали x_0 – [0, 1] і x_0 – [2.5, 3]. Розв'язок рівняння наведено у лістингу 6. При виборі початкового наближення потрібно бути уважним, особливо в тих випадках, коли рівняння містить кілька коренів. Може виявитися, що при встановленому користувачем наближенні x_0 буде визначений не той корінь, який очікувався. У нашому прикладі незначна зміна

x_0 призвела до іншого розв'язку: при x_0=1.9 функція FindRoot знайшла корінь x =0.258755, а вже при x_0=2 – x=2.94964.

```
In[59]:= F := 3^x - 9 x + 1 == 0
         FindRoot[F, {x, 0}]

Out[60]= {x → 0.258755}

In[61]:= FindRoot[F, {x, 3}]

Out[61]= {x → 2.94964}

In[62]:= FindRoot[F, {x, 1.9}]

Out[62]= {x → 0.258755}

In[63]:= FindRoot[F, {x, 2}]

Out[63]= {x → 2.94964}

In[64]:= FindRoot[F, {x, -50}]

Out[64]= {x → 0.258755}

In[65]:= FindRoot[F, {x, 20}]

Out[65]= {x → 2.94964}
```

Лістинг 6

Методи розв'язування систем рівнянь в системі Mathematica

Система Mathematica має багаті можливості розв'язування систем алгебраїчних рівнянь. Вбудовані функції дозволяють вирішувати системи лінійних та нелінійних рівнянь в аналітичному і чисельному вигляді. Дають можливість досить ефективно і оригінально перевіряти достовірність результатів. У процесі функціонування система видає коментарі, що дозволяють користувачеві приймати рішення щодо отриманих відповідей. Основними функціями розв'язання систем рівнянь є: Solve[F,X], Solve[F,X,Y], N[Solve[F,X]], FindRoot[F,X]. Розглянемо ці функції, опишемо технологію їх реалізації, наведемо приклади і задачі для самостійного розв'язання.

Функція Solve[F,X]

Функція Solve[F,X] дозволяє розв'язувати системи лінійних та нелінійних рівнянь в аналітичному вигляді. Вона записується наступним чином:

Solve[{f_1,f_2,...}, {x_1,x_2,...}]

де f_i – *i*-е рівняння, представлене в довільному вигляді, x_i – *i*-е невідоме.

Рівняння f_1, f_2, … можуть також представлятися через об'єднуючий знак &&. Приклади представлення функції Solve[F,X]:

Solve[{a*x^2+y==b,x+2*y==a+b},{x,y}]

Solve[a*x^2+y==b&&x+2*y==a+b,{x,y}]

При введенні рівнянь знак множення (*) можна замінити натисканням клавіші <Пробіл>. При вирішенні практичних завдань буває зручно, а в ряді випадків навіть доцільно, вводити рівняння окремо від функції Solve, присвоюючи їм імена, які потім вводити в функцію Solve замість рівнянь. Наприклад:

f_1=a*x^2+y==b; f_2=x+2*y==a+b

Solve[{f_1, f_2},{x,y}]

або

Solve[f_1&&f_2,{x,y}]

Така форма запису спрощує перевірку достовірності рішення системи рівнянь.

Системи лінійних алгебраїчних рівнянь

Методика розв'язування рівнянь:

1. Введення рівнянь з унікальним іменем, яке задається за допомогою знака присвоєння (=).
2. Запис функції Solve[{f_1,f_2,...},{x,y,...}] або Solve [f_1&&f_2&&...,{x,y,...}].
3. Перевірка достовірності розв'язку системи рівнянь.

Приклади розв'язання лінійних алгебраїчних рівнянь наведені в лістингу 7.

Необхідно розв'язати наступні системи лінійних рівнянь:

$$\begin{cases} x_1 + ax_2 - bx_3 = y_1; \\ (2+a)x_1 + x_2 + cx_3 = y_2; \\ ax_1 + bx_2 + cx_3 = y_3. \end{cases} \qquad \begin{cases} 3x_1 - 4x_2 + 2x_3 = 1; \\ x_1 + 7x_2 - 2x_3 = -4; \\ 2x_1 + 7x_2 + 3x_3 = 3. \end{cases} \qquad \begin{cases} x_1 + 7x_2 - x_3 = 3.5; \\ -1.6x_1 + 3.7x_2 = 12; \\ x_1 + 2x_2 + 5x_3 = 7.5. \end{cases}$$

```
In[66]:= f1 = x1 + a * x2 + b * x3 == y1

Out[66]= x1 + a x2 + b x3 == y1

In[67]:= f2 = (2 + a) * x1 + x2 + c * x3 == y2

Out[67]= (2 + a) x1 + x2 + c x3 == y2

In[68]:= f3 = a * x1 + b * x2 + c * x3 == y3

Out[68]= a x1 + b x2 + c x3 == y3

In[69]:= Solve[{f1, f2, f3}, {x1, x2, x3}]
```

Out[69]= $\left\{\left\{x1 \to -\frac{-c\ y1 + b\ c\ y1 - b^2\ y2 + a\ c\ y2 + b\ y3 - a\ c\ y3}{-a\ b + 2\ b^2 + a\ b^2 + c - 2\ a\ c - b\ c},\right.\right.$

$x2 \to -\frac{2\ c\ y1 + a\ b\ y2 - c\ y2 - 2\ b\ y3 - a\ b\ y3 + c\ y3}{-a\ b + 2\ b^2 + a\ b^2 + c - 2\ a\ c - b\ c},$

$\left.\left.x3 \to -\frac{a\ y1 - 2\ b\ y1 - a\ b\ y1 - a^2\ y2 + b\ y2 - y3 + 2\ a\ y3 + a^2\ y3}{-a\ b + 2\ b^2 + a\ b^2 + c - 2\ a\ c - b\ c}\right\}\right\}$

Лістинг 7 (Частина 1 з 2)

```
In[70]:= f3 = 3 x1 - 4 x2 + 2 x3 == 1

Out[70]= 3 x1 - 4 x2 + 2 x3 == 1

In[71]:= f4 = x1 + 7 x2 - 2 x3 == -4

Out[71]= x1 + 7 x2 - 2 x3 == -4

In[72]:= f5 = 2 x1 + 7 x2 + 3 x3 == 3

Out[72]= 2 x1 + 7 x2 + 3 x3 == 3

In[73]:= Solve[f3 && f4 && f5, {x1, x2, x3}]
```

Out[73]= $\left\{\left\{x1 \to -\frac{87}{119},\ x2 \to -\frac{3}{119},\ x3 \to \frac{184}{119}\right\}\right\}$

```
In[74]:= Solve[{x1 + 7 x2 - x3 == 3.5, -1.6 x1 + 3.7 x2 == 12,
           x1 + 2 x2 + 5 x3 == 7.5}, {x1, x2, x3}]

Out[74]= {{x1 -> -4.31818, x2 -> 1.37592, x3 -> 1.81327}}
```

Лістинг 7 (Частина 2 з 2)

З лістингу 7 видно, що система розв'язала першу систему рівнянь в аналітичному вигляді, другу – у вигляді точних значень невідомих, представлених в раціональній формі. Третя система рівнянь також розв'язана, але розв'язок представлено у вигляді дійсних чисел. Це пояснюється тим, що система рівнянь точного розв'язку не має. Таким чином, функція Solve може розв'язувати системи рівнянь також в чисельному вигляді. З прикладу видно, що розв'язок представляється у вигляді підстановок: x_1->, x_2->, x_3->. Це не дає можливості перевірити достовірність розв'язку, а також використовувати значення x_1, x_2, x_3 в подальших розрахунках.

Для перевірки достовірності розв'язку користувач повинен представити невідомі в явному вигляді з присвоєнням їм імені: x_1=-4.31818, x_2=1.37592, x3=1.81327, потім використовувати їх за призначенням. Отримати рішення в явному вигляді можна за допомогою виразу виду $\{x_1, x_2, x_3\}$/., яке ставиться перед функцією Solve: $\{x_1, x_2, x_3\}$/.Solve $\{f_1, f_2, f_3\}$, $\{x_1, x_2, x_3\}$]. Тепер x_1,x_2,x_3 можна використовувати за призначенням, у тому числі і для перевірки достовірності розв'язку системи рівнянь.

Системи нелінійних алгебраїчних рівнянь

Методика розв'язання систем нелінійних рівнянь та ж сама, що і лінійних. Це демонструється на прикладі розв’язку таких систем нелінійних рівнянь:

$$\begin{cases} x + ay = b; \\ x + by^2 = a + b. \end{cases} \qquad \begin{cases} xy = a; \\ x^2 + y^2 = ab. \end{cases}$$

Запис розв’язку та перевірки його достовірності наведено в лістингу 8.

```
In[75]:= f1 = x + a y == b
```

Out[75]= $x + a\,y == b$

```
In[76]:= f2 = x + b y^2 == a + b
```

Out[76]= $x + b\,y^2 == a + b$

```
In[77]:= w = Solve[{f1, f2}, {x, y}]
```

Out[77]= $$\left\{\left\{x \to \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b + \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right),\ y \to \frac{a - \sqrt{a}\,\sqrt{a+4\,b}}{2\,b}\right\},\right.$$
$$\left.\left\{x \to \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b - \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right),\ y \to \frac{a + \sqrt{a}\,\sqrt{a+4\,b}}{2\,b}\right\}\right\}$$

```
In[78]:= {f1, f2} /. w
```

Out[78]= $$\left\{\left\{\frac{a\left(a - \sqrt{a}\,\sqrt{a+4\,b}\right)}{2\,b} + \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b + \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right) == b,\right.\right.$$
$$\left.\frac{\left(a - \sqrt{a}\,\sqrt{a+4\,b}\right)^2}{4\,b} + \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b + \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right) == a + b\right\},$$
$$\left\{\frac{a\left(a + \sqrt{a}\,\sqrt{a+4\,b}\right)}{2\,b} + \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b - \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right) == b,\right.$$
$$\left.\left.\frac{\left(a + \sqrt{a}\,\sqrt{a+4\,b}\right)^2}{4\,b} + \frac{1}{2}\left(-\frac{a^2}{b} + 2\,b - \frac{a^{3/2}\sqrt{a+4\,b}}{b}\right) == a + b\right\}\right\}$$

```
In[79]:= Simplify[%]

Out[79]= {{True, True}, {True, True}}
```

Лістинг 8 (Частина 1 з 2)

```
In[80]:= f3 = x y == a
Out[80]= x y == a
In[81]:= f4 = x^2 + y^2 == a b
```

Out[81]= $x^2 + y^2 == a\, b$

```
In[82]:= r = Solve[{f3, f4}, {x, y}]
```

Out[82]=

$$\left\{\left\{x \to -\frac{\sqrt{a\,b - a\sqrt{-4+b^2}}}{\sqrt{2}},\ y \to \frac{-\sqrt{2}\,a\,b\sqrt{a\,b - a\sqrt{-4+b^2}} + \frac{\left(a\,b - a\sqrt{-4+b^2}\right)^{3/2}}{\sqrt{2}}}{2\,a}\right\},\right.$$

$$\left\{x \to \frac{\sqrt{a\,b - a\sqrt{-4+b^2}}}{\sqrt{2}},\ y \to \frac{2\sqrt{2}\,a\,b\sqrt{-a\left(-b+\sqrt{-4+b^2}\right)} - \sqrt{2}\left(-a\left(-b+\sqrt{-4+b^2}\right)\right)^{3/2}}{4\,a}\right\},$$

$$\left\{x \to -\frac{\sqrt{a\,b + a\sqrt{-4+b^2}}}{\sqrt{2}},\ y \to \frac{-2\sqrt{2}\,a\,b\sqrt{a\left(b+\sqrt{-4+b^2}\right)} + \sqrt{2}\left(a\left(b+\sqrt{-4+b^2}\right)\right)^{3/2}}{4\,a}\right\},$$

$$\left.\left\{x \to \frac{\sqrt{a\,b + a\sqrt{-4+b^2}}}{\sqrt{2}},\ y \to \frac{2\sqrt{2}\,a\,b\sqrt{a\left(b+\sqrt{-4+b^2}\right)} - \sqrt{2}\left(a\left(b+\sqrt{-4+b^2}\right)\right)^{3/2}}{4\,a}\right\}\right\}$$

```
In[83]:= Simplify[{f3, f4} /. r]
Out[83]= {{True, True}, {True, True}, {True, True}, {True, True}}
```

Лістинг 8 (Частина 2 з 2)

Функція Solve[F,X,Y]

Функція Solve[F,X,Y], так само, як і функція Solve[F,X], дозволяє розв'язувати системи лінійних та нелінійних рівнянь в аналітичному вигляді, але тільки з обмеженням: розв'язування здійснюються по змінним X і виключаються по змінним Y. Наприклад, функція Solve[{x+2*y-a==3, 2*x+y^2+b==7}, x, y] визначить X і виключить з розв'язку Y. Використання функції Solve[F,X,Y] демонструється в лістингу 9 при вирішенні таких систем рівнянь:

$$\begin{cases} ax^2 + bxy = 1; \\ xy + c = 7. \end{cases} \qquad \begin{cases} 7a + 3b + c = 1; \\ -5a + 12b = 3; \\ a + b + 2c = -7. \end{cases}$$

Першу систему розв'язано відносно x, другу – відносно a, потім відносно a і b.

```
In[84]:= f1 = a x^2 + b x y == 1
Out[84]= a x^2 + b x y == 1

In[85]:= f2 = x y + c == 7
Out[85]= c + x y == 7

In[89]:= z = Solve[{f1, f2}, x, y]
```

$$\text{Out[89]=}\ \left\{\left\{x \to -\frac{\sqrt{1-7\,b+b\,c}}{\sqrt{a}}\right\}, \left\{x \to \frac{\sqrt{1-7\,b+b\,c}}{\sqrt{a}}\right\}\right\}$$

```
In[90]:= x /. z
```

$$\text{Out[90]=}\ \left\{-\frac{\sqrt{1-7\,b+b\,c}}{\sqrt{a}}, \frac{\sqrt{1-7\,b+b\,c}}{\sqrt{a}}\right\}$$

```
In[91]:= f3 = 7 a + 3 b + c == 1
Out[91]= 7 a + 3 b + c == 1

In[92]:= f4 = -5 a + 12 b == 3
Out[92]= -5 a + 12 b == 3

In[93]:= f5 = a + b + 2 c == -7
Out[93]= a + b + 2 c == -7

In[96]:= s = Solve[{f3, f4, f5}, {a, b, c}]
```

$$\text{Out[96]=}\ \left\{\left\{a \to \frac{93}{181},\ b \to \frac{84}{181},\ c \to -\frac{722}{181}\right\}\right\}$$

```
In[97]:= s1 = Solve[{f3, f4, f5}, a, {b, c}]
```

$$\text{Out[97]=}\ \left\{\left\{a \to \frac{93}{181}\right\}\right\}$$

```
In[98]:= a /. s1
```

$$\text{Out[98]=}\ \left\{\frac{93}{181}\right\}$$

```
In[99]:= s2 = Solve[{f3, f4, f5}, {a, b}, c]
```

$$\text{Out[99]=}\ \left\{\left\{a \to \frac{93}{181},\ b \to \frac{84}{181}\right\}\right\}$$

```
In[100]:= {a, b} /. s2
```

$$\text{Out[100]=}\ \left\{\left\{\frac{93}{181},\ \frac{84}{181}\right\}\right\}$$

Лістинг 9

Функція NSolve[F,X]

Функція NSolve[F,X] дозволяє розв'язувати системи лінійних та нелінійних рівнянь в чисельному вигляді. Вона записується наступним чином:

Solve[{f_1, f_2,...}, {x_1, x_2,...}],

де f_i – *i*-е рівняння, представлене в довільному вигляді, x_i – *i*-е невідоме. Рівняння f_1, f_2, … можуть також бути представлені через об'єднуючий знак &&. Методика розв'язання систем рівнянь за допомогою функції NSolve[F,X] практично не відрізняється від технології вирішення за допомогою функції Solve{F,X}. За її допомогою в лістингу 10 вирішуються наступні системи рівнянь:

$$\begin{cases} 2x_1 + 7x_2 - x_3 = 5; \\ x_1 - 2x_2 + 5x_3 = 2; \\ 4x_1 + x_2 + 3x_3 = -7. \end{cases} \qquad \begin{cases} 2y + 3x^2 = 5; \\ x + 7y^2 = 7.5. \end{cases} \qquad \begin{cases} \sin x_1 - x_2 = 1.3 \\ \cos x_2 - x_1 = -0.82. \end{cases}$$

З лістингу 10 видно, що функція NSolve[F,X] не вирішила останню систему рівнянь. Це пояснюється тим, що система складається з трансцендентних рівнянь. Для її вирішення необхідні методи, які нереалізовані у функції NSolve[F,X].

```
In[102]:= f1 = 2 x1 + 7 x2 - x3 == 5
Out[102]= 2 x1 + 7 x2 - x3 == 5

In[103]:= f2 = x1 - 2 x2 + 5 x3 == 2
Out[103]= x1 - 2 x2 + 5 x3 == 2

In[104]:= f3 = 4 x1 + x2 + 3 x3 == -7
Out[104]= 4 x1 + x2 + 3 x3 == -7

In[105]:= s = NSolve[{f1, f2, f3}, {x1, x2, x3}]
Out[105]= {{x1 → -3.75, x2 → 2.06818, x3 → 1.97727}}

In[106]:= {f1, f2, f3} /. s
Out[106]= {{True, True, True}}

In[107]:= f1 = 2 y + 3 x^2 == 5
Out[107]= 3 x^2 + 2 y == 5

In[108]:= f2 = x + 7 y^2 == 7.5
Out[108]= x + 7 y^2 == 7.5

In[109]:= r = NSolve[{f1, f2}, {x, y}]
Out[109]= {{x → -1.55724, y → -1.13749}, {x → 1.01235, y → 0.962708},
           {x → 1.51106, y → -0.924966}, {x → -0.966177, y → 1.09975}}

In[110]:= {f1, f2} /. r
Out[110]= {{True, True}, {True, True}, {True, True}, {True, True}}

In[111]:= f3 = Sin[x1] - x2 == 1.3
Out[111]= -x2 + Sin[x1] == 1.3

In[112]:= f4 = Cos[x2] - x1 == -0.82
Out[112]= -x1 + Cos[x2] == -0.82

In[113]:= r1 = NSolve[{f3, f4}, {x1, x2}]
Out[113]= NSolve[{-x2 + Sin[x1] == 1.3, -x1 + Cos[x2] == -0.82}, {x1, x2}]
```

Лістинг 10

Функція FindRoot[F,{X,x0}]

Функція FindRoot[F, {X, x0}] розв'язує системи лінійних та нелінійних рівнянь чисельними методами ітерацій. Для її реалізації необхідно знати початкові наближення невідомих. Функція має вигляд:

FindRoot [{f_1, f_2, ...}, {x_1, x_{10}}, {x_2, x_{20}}, ...],

Де f_i – i-е рівняння, представлене в довільному вигляді, x_i – i-е невідоме, x_{i0} – початкове наближення i-го невідомого.

Рівняння f_1, f_2, … можуть також представлятися через об'єднувальний знак &&. Методика розв'язання систем рівнянь за допомогою функції FindRoot[F, {X, x0}] істотно відрізняється від технології розв'язання рівнянь за допомогою функції NSolve[F,X]. Відмінність полягає в необхідності визначення початкових наближень. Приклади розв'язку систем рівнянь функцією FindRoot[F,{X,x0}] наведено в лістингу 11. В лістингу 11 розв’язуються такі системи рівнянь:

$$\begin{cases} \sin x_1 - x_2 = 1.3; \\ \cos x_2 - x_1 = -0.82; \\ x_{10} = 1.8; \\ x_{20} = -0.35. \end{cases} \qquad \begin{cases} tg(y_1 y_2 + 0.2) = y_1^2; \\ 0.5 y_1^2 + 2 y_2^2 = 1; \\ y_{10} = 0.9; \\ y_{20} = 0.5. \end{cases}$$

```
In[114]:= f1 = Sin[x1] - x2 == 1.3

Out[114]= -x2 + Sin[x1] == 1.3

In[115]:= f2 = Cos[x2] - x1 == -0.82

Out[115]= -x1 + Cos[x2] == -0.82

In[116]:= r = FindRoot[{f1, f2}, {x1, 1.8}, {x2, -0.35}]

Out[116]= {x1 → 1.76935, x2 → -0.319646}

In[117]:= {f1, f2} /. r

Out[117]= {True, True}

In[118]:= x1 := 1.76935; x2 := -0.319646
          Sin[x1] - x2

Out[119]= 1.3

In[120]:= Cos[x2] - x1

Out[120]= -0.820003

In[121]:= {x1, x2} /. r

Out[121]= {1.76935, -0.319646}
```

Лістинг 11 (Частина 1 з 2)

```
In[122]:= f3 = Tan[y1 y2 + 0.2] == y1^2
Out[122]= Tan[0.2 + y1 y2] == y1^2

In[123]:= f4 = 0.5 y1^2 + 2 y2^2 == 1
Out[123]= 0.5 y1^2 + 2 y2^2 == 1

In[124]:= z = FindRoot[{f3, f4}, {y1, 0.9}, {y2, 0.5}]
Out[124]= {y1 → 0.910994, y2 → 0.540854}

In[125]:= {y1, y2} /. z
Out[125]= {0.910994, 0.540854}

In[126]:= y1 := 0.910994; y2 := 0.540854
          Tan[y1 y2 + 0.2]
Out[127]= 0.82991

In[128]:= y1^2
Out[128]= 0.82991

In[129]:= 0.5 y1^2 + 2 y2^2
Out[129]= 1.
```

Лістинг 11 (Частина 2 з 2)

Функція Eliminate[F,x]

Функція Eliminate[F,x] призначена для скорочення числа рівнянь системи шляхом виключення заданих змінних x. Вона має вигляд:

Eliminate[{f_1, f_2, ...}, {x_1, x_2, …}],

де f_i – *i*-е рівняння, представлене в довільному вигляді,

x_i – *i*-е невідоме, яке підлягає виключенню.

Рівняння f_1, f_2, ... можуть також представлятися через об'єднувальний знак &&. Ця функція здійснює перетворення вихідної системи рівнянь так, що число рівнянь і змінних скорочується. Граничним є одне рівняння з одним невідомим. Приклад використання функції Eliminate наведено в лістингу 12, в якому система з трьох рівнянь почергово зводиться до системи з двох рівнянь, а потім до одного рівняння.

$$\begin{cases} x^2 + ay^2 - z = 2; \\ x + y - bz = 7; \\ 2x + 3y + 9z = 1. \end{cases}$$

```
In[135]:= f1 = x^2 + a y^2 - z == 2
Out[135]= x^2 + a y^2 - z == 2

In[136]:= f2 = x + y - b z == 7
Out[136]= x + y - b z == 7

In[137]:= f3 = 2 x + 3 y + 9 z == 1
Out[137]= 2 x + 3 y + 9 z == 1

In[138]:= Eliminate[{f1, f2, f3}, z]
Out[138]= 9 a y^2 == 19 - 2 x - 9 x^2 - 3 y && b (-1 + 2 x + 3 y) == 63 - 9 x - 9 y

In[139]:= Eliminate[{f1, f2, f3}, {y, z}]
Out[139]= b^2 (-18 + a - 4 a x + 9 x^2 + 4 a x^2) +
            b (-48 + 126 a - 3 x - 270 a x + 54 x^2 + 36 a x^2) ==
           -18 - 3969 a + 9 x + 1134 a x - 81 x^2 - 81 a x^2

In[140]:= FullSimplify[Out[139]]
Out[140]= a (63 + b - 9 x - 2 b x)^2 + 3 (3 + b) (2 + x (-1 + 9 x) + 3 b (-2 + x^2)) == 0
```

Лістинг 12

Матричні методи розв'язання систем лінійних рівнянь

Систему лінійних алгебраїчних рівнянь можна представити в наступному вигляді: A*X=B, де A – матриця коефіцієнтів, X – вектор невідомих, B – вектор вільних членів (правих частин) системи рівнянь. Методика розв'язання рівнянь в системі Mathematica проста і полягає в наступному:

1. Введення матриці коефіцієнтів з присвоєнням їй імені, наприклад A.
2. Введення вектора невідомих з ім'ям X.
3. Введення вектора вільних членів з ім'ям *B*.
4. Створення виразу *z = A. X ==B.*
5. Введення функції *Solve [z, X]*.

Приклад розв’язання системи лінійних рівнянь

$$\begin{cases} 2x_1 + 3x_2 - 7x_3 = 1; \\ -3x_1 + x_2 + 5x_3 = 7.5; \\ 5x_1 + 3x_3 = 2.5, \end{cases}$$

матричним методом показана у лістингу 13.

```
In[141]:= A = {{2, 3, -7}, {-3, 1, 5}, {5, 0, 3}}
Out[141]= {{2, 3, -7}, {-3, 1, 5}, {5, 0, 3}}

In[142]:= X = {x1, x2, x3}
Out[142]= {x1, x2, x3}

In[143]:= B = {1, 7.5, 2.5}
Out[143]= {1, 7.5, 2.5}

In[144]:= Z = A.X == B
Out[144]= {2 x1 + 3 x2 - 7 x3, -3 x1 + x2 + 5 x3, 5 x1 + 3 x3} == {1, 7.5, 2.5}

In[145]:= Solve[%]
Out[145]= {{x1 → -0.0664336, x2 → 2.58042, x3 → 0.944056}}

In[146]:= {{2, 3, -7}, {-3, 1, 5}, {5, 0, 3}}.{x1, x2, x3} ==
           {1, 7.5, 2.5}
Out[146]= {2 x1 + 3 x2 - 7 x3, -3 x1 + x2 + 5 x3, 5 x1 + 3 x3} == {1, 7.5, 2.5}

In[147]:= Solve[%]
Out[147]= {{x1 → -0.0664336, x2 → 2.58042, x3 → 0.944056}}
```

Лістинг 13

Крім наведеного існують в системі Mathematica наступні два матричні способи розв'язання систем алгебраїчних рівнянь.

Спосіб 1. Визначення вектора невідомих X за формулою: $X=A^{-1}B$.

При цьому операція множення записується функцією *Dot*, а операція інвертування матриці *A*-функцією *Inverse*. Тоді рішення записується в наступному вигляді:

X: = Dot [Inverse [A], B]

Спосіб 2. Застосування функції *LinearSolve*.

Функція *LinearSolve* записується в наступному вигляді:

X: = LinearSolve [A, B]

У лістингу 14 наведено приклад вирішення системи лінійних рівнянь

$$\begin{cases} a_1x_1 + 2x_2 - 3x_3 = b_1; \\ -7x_1 + cx_2 + x_3 = b_2; \\ x_1 + x_2 + dx_3 = b_3, \end{cases}$$

двома способами.

Особливі випадки вирішення систем рівнянь

Система рівнянь може мати число розв’язків, рівне числу невідомих, така система називається *сумісною*. Якщо число рівнянь нескінченно велике, то систему називають *сумісною* та *невизначеною*. Якщо ж система не має жодного розв'язку, то її називають *несумісною*.

```
In[148]:= A = {{a, 2, -3}, {-7, c, 1}, {1, 1, d}}
Out[148]= {{a, 2, -3}, {-7, c, 1}, {1, 1, d}}

In[149]:= B = {b1, b2, b3}
Out[149]= {b1, b2, b3}

In[150]:= X = Dot[Inverse[A], B]
```

$$\text{Out[150]=}\left\{\frac{b3\,(2+3c)}{23-a+3c+14d+acd}+\frac{b2\,(-3-2d)}{23-a+3c+14d+acd}+\frac{b1\,(-1+cd)}{23-a+3c+14d+acd},\ \frac{(21-a)\,b3}{23-a+3c+14d+acd}+\frac{b1\,(1+7d)}{23-a+3c+14d+acd}+\frac{b2\,(3+ad)}{23-a+3c+14d+acd},\ \frac{(2-a)\,b2}{23-a+3c+14d+acd}+\frac{b1\,(-7-c)}{23-a+3c+14d+acd}+\frac{b3\,(14+ac)}{23-a+3c+14d+acd}\right\}$$

```
In[151]:= Simplify[%]
```

$$\text{Out[151]=}\left\{\frac{b3\,(2+3c)-b2\,(3+2d)+b1\,(-1+cd)}{23+3c+14d+a\,(-1+cd)},\ \frac{b1+3\,b2+21\,b3-a\,b3+7\,b1\,d+a\,b2\,d}{23-a+3c+14d+acd},\ \frac{-(-2+a)\,b2-b1\,(7+c)+b3\,(14+ac)}{23+3c+14d+a\,(-1+cd)}\right\}$$

```
In[152]:= X = LinearSolve[A, B]
```

$$\text{Out[152]=}\left\{\frac{-b1-3\,b2+2\,b3+3\,b3\,c-2\,b2\,d+b1\,c\,d}{23-a+3c+14d+acd},\ \frac{b1+3\,b2+21\,b3-a\,b3+7\,b1\,d+a\,b2\,d}{23-a+3c+14d+acd},\ \frac{-7\,b1+2\,b2-a\,b2+14\,b3-b1\,c+a\,b3\,c}{23-a+3c+14d+acd}\right\}$$

Лістинг 14

Дві наступні системи рівнянь демонструють особливі випадки при розв'язку систем лінійних рівнянь:

$$\begin{cases} x_1+2x_2-x_3=1; \\ 2x_1+4x_2-2x_3=2; \\ 3x_1+2x_2+3x_3=5. \end{cases} \qquad \begin{cases} x_1+2x_2-x_3=1; \\ 2x_1+4x_2-2x_3=2; \\ 3x_1+6x_2-3x_3=3. \end{cases}$$

Їх розв'язок наведено у лістингу 15.

```
In[157]:= Solve[{x1 + 2 x2 - x3 == 1, 2 x1 + 4 x2 - 2 x3 == 2,
            3 x1 + 2 x2 + 3 x3 == 5}, {x1, x2, x3}]

Out[157]= {{x2 → 1 - 3 x1/4, x3 → 1 - x1/2}}

In[163]:= Solve[{x1 + 2 x2 - x3 == 1, 2 x1 + 4 x2 - 2 x3 == 2, 3 x1 + 6 x2 - 3 x3},
           {x1, x2, x3}]

Out[163]= Solve[{x1 + 2 x2 - x3 == 1, 2 x1 + 4 x2 - 2 x3 == 2, 3 x1 + 6 x2 - 3 x3},
           {x1, x2, x3}]

In[164]:= Det[{{1, 2, -1}, {2, 4, -2}, {3, 2, 3}}]

Out[164]= 0

In[165]:= Det[{{1, 2, -1}, {2, 4, -2}, {3, 6, -3}}]

Out[165]= 0
```

Лістинг 15

З лістингу 15 видно, що перша система рівнянь має розв'язок: $x_2 = 1 - \frac{3x_1}{4}, x_3 = 1 - \frac{x_1}{2}$, тобто має нескінченне число розв'язків (при будь-якому значенні x_1). Система сумісна, але невизначена. Друга система також є сумісною і невизначеною, маючи розв'язок: $x_3 = -1 + x_1 + 2x_2$, підставляючи значення x_1, x_2. Зазначимо, що в обох випадках головний визначник системи дорівнює нулю.

Хід роботи

1. Знайти розв'язок трансцендентного рівняння в аналітичному і чисельному виді згідно з варіантом, вказаним у табл.1.

Таблиця 1. Трансцендентне рівняння

Вар.	Рівняння	Вар.	Рівняння	Вар.	Рівняння
1,11	$3x^2 - e^x = 0$	4,14	$9 - x^2 = e^x$	7,17	$x \cdot tg(x/2) = \mathbf{sin}(x/2)$
2,12	$\mathbf{lg}\, x = x - 5$	5,15	$\mathbf{sin}(2x) = 6 - x^2$	8,18	$2^x = x + 5$
3,13	$2^x = \mathbf{lg}\, x + x^5 + 40$	6,16	$\mathbf{ln}(2 + x) = 0.4x^3$	9,19	$\mathbf{log}_3(5x - 6) = 2\sqrt{10 - 3x}$
10,20	$\mathbf{log}_2\, x = x^2 - x$				

2. Знайти корені поліноміального рівняння в аналітичному і чисельному виді згідно з варіантом, вказаним у табл.2.

Таблиця 2. Поліноміальне рівняння

Вар.	Рівняння	Вар.	Рівняння	Вар.	Рівняння
1,11	$x^4 + x^2 + 1 = 0$	4,14	$x^5 - 3x^2 + x - 12 = 0$	7,17	$x^7 - 3x^5 + 2x - 4 = 0$
2,12	$x^5 - 4x^3 + 2x^2 + 1 = 0$	5,15	$x^4 - 6x^3 + 3x - 2 = 0$	8,18	$x^5 - 8x^4 + 4x^2 - 6 = 0$

3,13	$x^6-4x^2+3x^3+x-25=0$	6,16	$x^8-1=0$	9,19	$x^4-6x^3+2x-8=0$
				10,20	$x^5-3x^4+3x^3-1=0$

3. Розв'зати систему лінійних рівнянь згідно з варіантом, вказаним у табл. 3 за допомогою функцій Solve [F,X], Solve [F,X,Y], NSolve[F,X] і матричним методом. Пояснити особливості використання цих функцій.

Таблиця 3. Система лінійних рівнянь

Вар.	Система рівнянь	Вар.	Система рівнянь
1,11	$\begin{cases}1.5x_1-0.8x_2+4.25x_3=5.1;\\1.2x_1+7.18x_2-3.2x_3=4.2;\\0.5x_1-1.5x_2+7.1x_3=-1.2.\end{cases}$	2,12	$\begin{cases}6.7x_1-0.6x_2+0.83x_3=6.8;\\0.8x_1+1.1x_2+7.2x_3=5.2;\\1.2x_1+5.4x_2-0.54x_3=-3.2.\end{cases}$
3,13	$\begin{cases}-1.32x_1+2.15x_2+7.6x_3=-1.4;\\2.62x_1+6.1x_2-4.12x_3=5.6;\\8.3x_1-2.84x_2-1.5x_3=-6.5.\end{cases}$	4,14	$\begin{cases}0.51x_1-10x_2-3.62x_3=-2.05;\\3.09x_1+1.23x_2-4.64x_3=-5.6;\\3.2x_1-2.31x_2-8.4x_3=6.1.\end{cases}$
5,15	$\begin{cases}7.12x_1-6.66x_2+2.6x_3=-3.1;\\-1.7x_1+6.5x_2-0.87x_3=2.85;\\0.65x_1+0.87x_2-8.7x_3=5.56.\end{cases}$	6,16	$\begin{cases}6.4x_1-0.73x_2+2.1x_3=3.8;\\-1.07x_1+3.8x_2-1.5x_3=-1.2;\\2.7x_1-3.1x_2+4.2x_3=-7.5.\end{cases}$
7,17	$\begin{cases}9.21x_1-1.84x_2+0.7x_3=-3.2;\\-6.17x_1+8.5x_2-2.87x_3=-3.75;\\0.7x_1+0.87x_2-8.7x_3=2.64.\end{cases}$	8,18	$\begin{cases}4.3x_1-1.2x_2+10.3x_3=4.2;\\0.21x_1+6.2x_2+3.54x_3=5.1;\\-0.31x_1-0.52x_2+3.6x_3=-2.1.\end{cases}$
9,19	$\begin{cases}6.9x_1-2.3x_2+1.21x_3=3.1;\\x_1+2.3x_2-3.4x_3=-2.3;\\0.21x_1-0.43x_2+6.3x_3=3.6.\end{cases}$	10,20	$\begin{cases}12.4x_1-0.56x_2+4.2x_3=6.3;\\-0.65x_1+4.4x_2+1.5x_3=1.5;\\1.5x_1+2.1x_2-2.8x_3=1.7.\end{cases}$

4. Розв'зати систему нелінійних рівнянь згідно з варіантом, вказаним у табл. 4 за допомогою функції FindRoot[F,{X,x0}].

Таблиця 4. Система лінійних рівнянь

Вар.	Система рівнянь	Початкові наближення
1,11	$\begin{cases}\sin(x_1+x_2)-1.2x_1=0.1;\\x_1^{\,2}+x_2^{\,2}=1.\end{cases}$	$x_1=0.74, x_2=0.67.$
2,12	$\begin{cases}tg(x_1x_2+0.2)=x_1^{\,2};\\0.6x_1^{\,2}+2x_2^{\,2}=1.\end{cases}$	$x_1=0.88, x_2=0.52.$
3,13	$\begin{cases}\sin x_1+2\sin x_2=1;\\2\sin 3x_1^{\,2}+3\sin 3x_2^{\,2}=0.3.\end{cases}$	$x_1=1.08, x_2=0.06.$

4,14	$\begin{cases} tg(x_1 - x_2) - 4x_1 = 0; \\ x_1^2 + 2x_2^2 = 1. \end{cases}$	$x_1 = -0.5, x_2 = 0.6.$
5,15	$\begin{cases} x_1^4 + x_2^2 - 3 = 0; \\ x_1^3 + x_2^3 - 4 = 0. \end{cases}$	$x_1 = 0.95, x_2 = 1.4.$
6,16	$\begin{cases} \sin x_1 - x_2 = 1.3; \\ \cos x_2 - x_1 = -0.82. \end{cases}$	$x_1 = 1.8, x_2 = -0.35.$
7,17	$\begin{cases} x_1^3 + x_2^3 - 6x_1 + 3 = 0; \\ x_1^3 + x_2^3 - 6x_2 = 2. \end{cases}$	$x_1 = 0.52, x_2 = -0.37.$
8,18	$\begin{cases} x_2 + e^{x_1 - x_2} = 0; \\ x_1 + e^{x_1 + x_2} = 0. \end{cases}$	$x_1 = -0.7, x_2 = -0.35.$
9,19	$\begin{cases} \sin(x_2 + 1) - x_1 = 1.2; \\ 2x_2 + \cos x_1 = 2. \end{cases}$	$x_1 = 0.1, x_2 = -0.1.$
10,20	$\begin{cases} \cos(x_2 - 1) + x_1 = 0.5; \\ x_2 - \cos x_1 = 3. \end{cases}$	$x_1 = 0, x_2 = \pi.$

5. За допомогою функції Eliminate[F,x] з системи лінійних рівнянь згідно з табл. 5 виразити аналітичний вираз вказаної змінної.

Таблиця 5. Система лінійних рівнянь

Вар.	Система рівнянь	Змінна	Вар.	Система рівнянь	Змінна
1,11	$\begin{cases} 1.2x_1 - 1.06x_2 - 6.7x_3 + 5.3x_4 = 2.12; \\ 4.2x_1 - 6.3x_2 - 0.9x_3 - 1.7x_4 = -1.1; \\ 0.6x_1 + 6.8x_2 + 0.82x_3 + 1.3x_4 = 0.83. \end{cases}$	x_1	2,12	$\begin{cases} 9.7x_1 + 0.35x_2 - 1.84x_3 + 0.6x_4 = 2.15; \\ 4.64x_1 - 7.1x_2 - 4.3x_3 + 2.2x_4 = 1.5; \\ 0.32x_1 + 0.348x_2 - 3.3x_3 - 1.7x_4 = -3.1. \end{cases}$	x_3
3,13	$\begin{cases} 6.5x_1 - 2.34x_2 + 1.4x_3 - 1.5x_4 = 2.8; \\ 0.5x_1 + 7.3x_2 - 2.4x_3 + 2.7x_4 = -3.8; \\ 8.6x_1 + 0.34x_2 - 6.4x_3 - 1.7x_4 = 0.64. \end{cases}$	x_2	4,14	$\begin{cases} 2.8x_1 + 4.3x_2 - 3.7x_3 + 0.8x_4 = 5.1; \\ -0.45x_1 - 8.24x_2 + 4.8x_3 + 5.2x_4 = 5.4; \\ 0.54x_1 + 2.3x_2 + 3.7x_3 - 1.7x_4 = 1.54. \end{cases}$	x_1
5,15	$\begin{cases} 6x_1 + 0.13x_2 - 0.67x_3 - 0.8x_4 = 1.9; \\ 3.8x_1 + 1.25x_2 - 4.3x_3 + 3.2x_4 = 6.4; \\ 0.38x_1 - 0.64x_2 + 3.2x_3 + 2.3x_4 = 5.4. \end{cases}$	x_3	6,16	$\begin{cases} 1.5x_1 - 2.6x_2 + 7x_3 - 0.8x_4 = -11.2; \\ 6.6x_1 + 1.3x_2 - 1.24x_3 + 0.9x_4 = 5.3; \\ 0.85x_1 - 8.4x_2 + 4.7x_3 + 2.7x_4 = 1.6. \end{cases}$	x_2
7,17	$\begin{cases} 6.2x_1 - 0.52x_2 + 2.3x_3 - 8.7x_4 = -1.8; \\ -4.2x_1 + 3.4x_2 - 0.5x_3 - 5.4x_4 = 0.7; \\ 0.2x_1 + 0.8x_2 + 3.6x_3 + 9.1x_4 = 3.2. \end{cases}$	x_1	8,18	$\begin{cases} 0.63x_1 - 0.54x_2 + 1.7x_3 - 4.7x_4 = 3.6; \\ 0.65x_1 + 4.4x_2 + 0.15x_3 + 2.3x_4 = 2.3; \\ 1.5x_1 + 0.2x_2 + 4.1x_3 - 6.7x_4 = 2.8. \end{cases}$	x_3
9,19	$\begin{cases} 8.4x_1 - 0.25x_2 + 3.1x_3 + 5.2x_4 = -5.7; \\ -0.3x_1 + 6.1x_2 - 1.54x_3 - 6.9x_4 = 3.3; \\ -6.8x_1 + 1.2x_2 - 7x_3 - 8.5x_4 = 4.5. \end{cases}$	x_2	10,20	$\begin{cases} 12x_1 + 4.2x_2 - 0.8x_3 - 3.5x_4 = -5.4; \\ -4.1x_1 + 2.2x_2 - 0.16x_3 - 4.5x_4 = 1.6; \\ -1.6x_1 - 4.3x_2 + 8.4x_3 + 3.1x_4 = 12.2. \end{cases}$	x_1

Контрольні запитання

1. Перелічіть функції, призначені для вирішення рівнянь.
2. Перелічіть функції, які використовують для вирішення поліноміальних рівнянь.
3. Перелічіть функції, які використовують для вирішення трансцендентних рівнянь.
4. Вкажіть особливості використання функцій, які використовують для отримання аналітичного і чисельного розв’язку рівнянь.
5. Вкажіть функцію, яку використовують для скорочення рівнянь системи.
6. Перелічіть модифікації функції Solve[], призначені для розв’язання систем рівнянь, і перелічіть особливості їх використання.
7. Опишіть методику розв’язку системи рівнянь матричним методом.
8. Опишіть особливі випадки вирішення систем рівнянь.
9. Опишіть метод локалізації коренів рівняння.
10. Вкажіть, яким має бути значення параметра початкового наближення у функції FindRoot[].

Лабораторна робота № 5. Функції математичного аналізу

Мета: вивчення функцій математичного аналізу, реалізованих в програмі Mathematica.

Короткі теоретичні відомості

Обчислення сум і добутків рядів

Обчислення сум рядів може здійснюватись в аналітичному або числовому вигляді. Для обчислення сум в аналітичному вигляді використовується функція Sum. Існують наступні формати запису функції Sum:

- Sum[f_i, {i, i_{max}}];
- Sum[f_i, {i, i_{min}, i_{max}}];
- Sum[f_i, {i, i_{min}, i_{max}, Δi}];
- Sum[$f_{i,j,..}$ {i, i_{min}, i_{max}}, {j, j_{min}, j_{max}}],

де f – елемент сумування,
i, j – змінні сумування,
i_{min}, i_{max} – елементи сумування,
Δi – крок зміни аргументу i.

Якщо необхідно розрахувати суму членів ряду, який представлений аналітичною функцією, від 1 до n, необхідно використовувати функцію Sum[f_i, {i, i_{max}}]. В лістингу 1 наведено використання функції для обчислення рядів 1/n, 1/n^2, 1/n^3.

```
In[168]:= Sum[n, {n, n}]
Out[168]= 1/2 n (1 + n)

In[169]:= Sum[n^2, {n, n}]
Out[169]= 1/6 n (1 + n) (1 + 2 n)

In[170]:= Sum[n^3, {n, n}]
Out[170]= 1/4 n^2 (1 + n)^2

In[171]:= Sum[n, {n, 1000}]
Out[171]= 500 500

In[172]:= Sum[n, {n, 1 000 000}]
Out[172]= 500 000 500 000
```

Лістинг 1. Використання функції Sum[f_i, {i, i_{max}}]

Функція Sum[f_i, {i, i_{min}, i_{max}}] обчислює суму значень функції f в діапазоні значень аргументу i_{min} ..i_{max} з кроком 1. При використанні функції Sum[f_i, {i, i_{min}, i_{max}, Δi}] крок зміни аргументу задається параметром Δi.

Функція Sum[$f_{i,j,..}$ {i, i_{min}, i_{max}}, {j, j_{min}, j_{max}}] обчислює суму за декількома змінними. Приклад її використання ілюструється при обчисленні наступних рядів: $\sum_{i=1}^{50}\sum_{j=1}^{10}(x_i^2+y_j^2)$, $\sum_{n=0}^{\infty}\sum_{m=0}^{\infty}\frac{x^n}{n!}\frac{y^m}{m!}$, $\sum_{n=1}^{3}\sum_{m=1}^{3}\frac{x^n}{n!}\frac{y^m}{m!}$ показано у лістингу 2.

```
In[175]:= Sum[x^2 + y^2, {x, 1, 50}, {y, 1, 10}]

Out[175]= 448 500

In[173]:= Sum[(x^n / n!) * (y^m / m!), {n, 0, Infinity}, {m, 0, Infinity}]

Out[173]= e^(x+y)

In[176]:= Sum[(x^n / n!) * (y^m / m!), {n, 1, 3}, {m, 1, 3}]
```

Out[176]= $x\,y+\frac{x^2 y}{2}+\frac{x^3 y}{6}+\frac{x\,y^2}{2}+\frac{x^2 y^2}{4}+\frac{x^3 y^2}{12}+\frac{x\,y^3}{6}+\frac{x^2 y^3}{12}+\frac{x^3 y^3}{36}$

Лістинг 2

Чисельне обчислення сум виконується функцією NSum, яка має такі ж модифікації, як і функція Sum.

Обчислення добутків здійснюється аналогічно сумуванню. Для цього використовуються функції:

- Product – для обчислення добутків в аналітичному і чисельному вигляді;
- NProduct - для обчислення добутків тільки в чисельному вигляді.

Обчислення границі функції

Обчислення границі функції в системі Mathematica здійснюється за допомогою функції Limit. Синтаксис її запису такий:

Limit[f(x), x->x_0].

Функція Limit має опцію Direction. Опція Direction вказує на напрям наближення до границі. Її запис має два варіанти:

Direction -> +1;

Direction -> -1.

Значення +1 свідчить про наближення до границі з лівого боку, -1 – з правого боку. Приклад використання функції Limit показано у лістингу 3.

```
In[177]:= Limit[(1 - Exp[-2 * t]) / Log[1 - 3 * t], t → 0]
```

Out[177]= $-\frac{2}{3}$

```
In[178]:= Limit[ArcTan[1 / x], x → 0, Direction → +1]
```

Out[178]= $-\frac{\pi}{2}$

```
In[179]:= Limit[ArcTan[1 / x], x → 0, Direction → -1]
```

Out[179]= $\frac{\pi}{2}$

Лістинг 3

Розклад функцій у степеневий ряд

Для розкладу в степеневий ряд в системі Mathematica використовуються наступні функції:

Series[f,{x,x0,n}] – розкладає функцію f в околі точки x = x_0 при використанні n членів ряду;

Series[f(x,y),{x,x0,nx}, {y,y0,ny}] – розкладає функцію f по двом змінним x і y в околі точки (x0, y0) з числом членів nx і ny відповідно.

Приклади використання функцій наведено у лістингу 4.

```
In[180]:= Series[Exp[x], {x, 0, 7}]
```

Out[180]= $1+x+\frac{x^2}{2}+\frac{x^3}{6}+\frac{x^4}{24}+\frac{x^5}{120}+\frac{x^6}{720}+\frac{x^7}{5040}+O[x]^8$

```
In[181]:= Series[Exp[x + y], {x, 0, 3}, {y, 0, 3}]
```

Out[181]= $\left(1+y+\frac{y^2}{2}+\frac{y^3}{6}+O[y]^4\right)+\left(1+y+\frac{y^2}{2}+\frac{y^3}{6}+O[y]^4\right)x+\left(\frac{1}{2}+\frac{y}{2}+\frac{y^2}{4}+\frac{y^3}{12}+O[y]^4\right)x^2+\left(\frac{1}{6}+\frac{y}{6}+\frac{y^2}{12}+\frac{y^3}{36}+O[y]^4\right)x^3+O[x]^4$

Лістинг 4

Розрахунок похідних функцій

Розрахунок похідних здійснюють за допомогою використання таких функцій:

D[f,x];

D[f,{x,n}];

D[f,x_1,x_2,…];

Dt[f,x];

Dt[f];

Derivative[n_1,n_2,..][f] – представляє собою похідну від f[{x_1, x_2, …}] взяту n_i разів по x_i.

Де f – дифереційована функція,

x – змінна диферецiювання,

x_1, x_2,.. – змінні диферецiювання,

n – порядок похідної.

Методи обчислення інтегралів

Аналітичні методи

Інтеграл в аналітичному вигляді обчислюється за допомогою наступних вбудованих функцій :

- $Integrate\,[f(x),x]$- обчислює невизначений інтеграл функції $f(x)$ за аргументом x ;

- $Integrate\,[\,f(x),\{x,x_н,x_к\}]$ - обчислює визначений інтеграл функції $f(x)$ за змінною x з нижньою $x_н$ і верхньою $x_к$ межами інтегрування. Межами інтегрування можуть бути символьні змінні, числа і навіть функції;

- $Integrate\,[\,f(x,y,\ldots),\{x,x_н,x_к\},\{y,y_н,y_k,\ldots\}]$ – Обчислює визначений інтеграл функції багатьох змінних $x,y,\ldots$ з межами інтегрування $x_н$, $x_к$, $y_н$, $y_к$.

Приклади обчислення інтегралів наведено у лістингу 5. Обчислюються такі інтеграли:

$$\int \frac{ax-1}{bx+1}dx, \int_a^b \frac{2+x}{x}dx, \int_a^b (1+2xy+4x^2y^2)\,dxdy\,.$$

```
In[194]:= f1 := (a * x - 1) / (b * x + 1)
          Integrate[f1, x]

Out[195]= (a b x - (a + b) Log[1 + b x]) / b^2

In[200]:= f2 := (2 + x) / x
          Integrate[f2, {x, a, b}]

          -a + b + 2 (-Log[a] + Log[b])

In[198]:= f3 := 1 + 2 * x * y + 4 * x^2 * y^2
          Integrate[f3, {x, a, b}, {y, a, b}]

Out[199]= (-a + b)^2 + 2 (-a^2/2 + b^2/2)^2 + 4 (-a^3/3 + b^3/3)^2
```

Лістинг 5

У попередньому прикладі межами інтегрування були символьні змінні a і b. У лістингу 6 показано обчислення інтегралів тих же функцій у випадку, коли межами інтегрування є числа : $x_H = 1$, $x_K = 5$, $y_H = 0$, $y_K = 10$.

```
In[202]:= Integrate[(2 + x) / x, {x, 1, 5}]

Out[202]= 4 + Log[25]

In[203]:= N[%]

Out[203]= 7.21888

In[204]:= Integrate[1 + 2 x y + 4 x^2 y^2, {x, 1, 5}, {y, 0, 10}]

Out[204]= 507160 / 9

In[205]:= N[%]

Out[205]= 56351.1
```

Лістинг 6

З лістингу 6 видно, що в даному випадку отримані у вигляді точних рішень. Числові значення інтегралів отримано за допомогою функції N(%).

Чисельні методи

Обчислення інтегралів у чисельному вигляді необхідно в наступних випадках:

- первісна не виражається через елементарні функції;
- підінтегральна функція задана у вигляді таблиці;
- аналітичний вираз первісної занадто складний.

Як приклад у лістингу 7 наведені обчислення невизначених і визначених інтегралів від функцій: $y(x) = x^{\frac{1}{x}}e^x$ и $y(x) = x^{20}e^{-x}$.

```
In[206]:= Integrate[x^(1/x) * Exp[x], x]
```

$$\text{Out[206]=} \int e^x x^{\frac{1}{x}} \, dx$$

```
In[207]:= NIntegrate[x^(1/x) * Exp[x], {x, 1, 5}]
Out[207]= 204.435

In[208]:= Integrate[x^20 Exp[-x], x]
```

$$\text{Out[208]=} e^{-x} (-2\,432\,902\,008\,176\,640\,000 - 2\,432\,902\,008\,176\,640\,000\, x - 1\,216\,451\,004\,088\,320\,000\, x^2 - 405\,483\,668\,029\,440\,000\, x^3 - 101\,370\,917\,007\,360\,000\, x^4 - 20\,274\,183\,401\,472\,000\, x^5 - 3\,379\,030\,566\,912\,000\, x^6 - 482\,718\,652\,416\,000\, x^7 - 60\,339\,831\,552\,000\, x^8 - 6\,704\,425\,728\,000\, x^9 - 670\,442\,572\,800\, x^{10} - 60\,949\,324\,800\, x^{11} - 5\,079\,110\,400\, x^{12} - 390\,700\,800\, x^{13} - 27\,907\,200\, x^{14} - 1\,860\,480\, x^{15} - 116\,280\, x^{16} - 6840\, x^{17} - 380\, x^{18} - 20\, x^{19} - x^{20})$$

```
In[209]:= NIntegrate[x^20 Exp[-x], {x, 1, 2}]
Out[209]= 14860.3
```

Лістинг 7

З лістингу 7 видно, що перший інтеграл аналітично не вирішується, а другий є занадто громіздким. В цих випадках результат обчислення інтегралів чисельними методами можна отримати за допомогою функції NIntegrate.

Формат функції NIntegrate є таким:

$NIntegrate(f(x), \{x, x_H, x_K\})$,

де *f(x)* - підінтегральна функція ;

x - аргумент підінтегральної функції ;

x_H, x_K - нижня і верхня межі інтегрування.

Методика використання функції NIntegrate не відрізняється від методики обчислення визначеного інтеграла в аналітичному вигляді.

Обчислення кратних інтегралів

Обчислення кратних інтегралів в системі Mathematica здійснюється шляхом багаторазового застосування функції Integrate при аналітичному інтегруванні або NIntegrate при чисельному інтегруванні. У лістингу 8 наведено приклади обчислення кратних інтегралів від функції $f(x) = \frac{x-1}{x+1}$ при різних варіантах меж інтегрування:

- символьних змінних від a до b;
- чисельних значеннях меж інтегрування від 0 до 2;
- меж інтегрування , заданих функціями: ln 2 і $e^{1.2}$.

```
Integrate[Integrate[(x - 1) / (x + 1), x], {x, a, b}]
(*∫_a^b ∫ (x-1) / (x+1) dx dx*)
```

$$\frac{1}{2}\left(-a^2+b^2-4\,(a-b-(1+a)\ \text{Log}[1+a]+(1+b)\ \text{Log}[1+b])\right);$$

```
In[212]:= (*∫_a^b ∫∫ (x-1) / (x+1) dx dx dx*)
          Integrate[Integrate[Integrate[(x - 1) / (x + 1), x], x],
           {x, a, b}]
```

$$-\frac{1}{6}\,a\left(6+9\,a+a^2\right)+\frac{1}{6}\,b\left(6+9\,b+b^2\right)+(1+a)^2\ \text{Log}[1+a]-(1+b)^2\ \text{Log}[1+b];$$

```
In[213]:= (*∫_0^2 ∫∫ (x-1) / (x+1) dx dx dx*)
          Integrate[Integrate[Integrate[(x - 1) / (x + 1), x], x],
           {x, 0, 2}]
```

Out[213]= $\frac{28}{3} - \text{Log}[19683]$

```
In[214]:= N[%]

Out[214]= -0.554177

In[215]:= (*∫_Log[2]^Exp[1.2] ∫∫ (x-1) / (x+1) dx dx dx*)
          Integrate[Integrate[Integrate[(x - 1) / (x + 1), x], x],
           {x, Log[2], Exp[1.2]}]

Out[215]= -1.31499
```

Лістинг 8

Обчислення невласних інтегралів

Система Mathematica дозволяє обчислювати інтеграли з нескінченними границями. При цьому використовуються ті ж функції, що й у випадку обчислення інтегралів з кінцевими межами. Значення нескінченності позначається або символом ∞ або службовою константою Infinity. Приклади обчислення невласних інтегралів наведено у лістину 9.

```
In[216]:= (*∫_0^∞ Sin[x]/x dx*)
          Integrate[Sin[x] / x, {x, 0, Infinity}]
```

$$\text{Out[216]=}\quad \frac{\pi}{2}$$

```
In[217]:= (*∫_0^∞ (Sinh[x]+Cosh[x])/Exp[x^2] dx*)
          Integrate[(Sinh[x] + Cosh[x]) / Exp[x^2], {x, 0, Infinity}]
```

$$\text{Out[217]=}\quad \frac{1}{2} e^{1/4} \sqrt{\pi} \left(1 + \text{Erf}\left[\frac{1}{2}\right]\right)$$

```
In[218]:= N[%]
Out[218]= 1.73023
```

```
In[219]:= (*∫_0^∞ (x+1)/(x^3+1) dx*)
          Integrate[(x + 1) / (x^3 + 1), {x, 0, Infinity}]
```

$$\text{Out[219]=}\quad \frac{4\pi}{3\sqrt{3}}$$

```
In[222]:= (*∫_-∞^∞ Exp[x]/(a+Exp[x])^2 dx*)
          Integrate[Exp[x] / (a + Exp[x])^2, {x, -Infinity, Infinity}]
```

$$\text{Out[222]=}\quad \text{ConditionalExpression}\left[\frac{1}{a},\ \text{Im}[a] \neq 0 \;||\; \text{Re}[a] == -1 \;||\; \text{Re}[a] \geq 0\right]$$

```
In[223]:= (*∫_0^∞ (x+1)/(2 x+1) dx*)
          Integrate[(x + 1) / (2 x + 1), {x, 0, Infinity}]
```

$$\text{Out[223]=}\quad \int_0^\infty \frac{1+x}{1+2x}\,dx$$

```
In[224]:= NIntegrate[1 / (2 x^2 + 1), {x, 0, Infinity}]
Out[224]= 1.11072
```

Лістинг 9

З лістингу 9 можна зробити наступні висновки:

- вирішення невласного інтеграла отримуємо в аналітичному вигляді. Для отримання чисельного значення інтеграла застосовується команда N [%];
- вираз первісної функції в аналітичному вигляді може бути складним. Для його спрощення слід скористатися функціями Simplify , Expand, Factor або їх прототипами;
- якщо інтеграл не має первісної, то результатом буде початковий вираз інтеграла.

Табличне інтегрування

Підінтегральна функція f(x) може бути задана у вигляді таблиці. Найчастіше це необхідно при проведенні експериментальних досліджень. У таких випадках обчислення інтеграла можна здійснити за формулами прямокутників, трапецій або парабол . Рішення можна отримати також шляхом інтерполяції функції f (x) з подальшим її інтегруванням. Система Mathematica після версії 6.0 змінило вбудовану функцію ListIntegrate на складену форму Integrate[Interpolation[]], що має наступні особливості використання:

- $Integrate[\text{Interpolation}[\{y_1, y_2, \dots, y_n\}][x], \{x, x_1, x_2\}];$
- $Integrate[\text{Interpolation}[\{y_1, y_2, \dots, y_n\}, \text{InterpolationOrder} \to k][x], \{x, x_1, x_2\}];$
- *Integrate*[Interpolation[data][x],{x,Min[x_c],Max[x_c]}];

У функціях використовуються наступні позначення:

- y_i - значення функції y = f(x) в x_i – му вузлі , 1 = 1 , 2 , ... , n;
- x – аргумент функції y = f(x).
- x_1, x_2 – кінцеві значення аргументу;
- k – порядок інтерполяції.
- data $= \{\{x_1, y_1\}, \dots, \{x_n, y_n\}\}$;
- $x_c = \text{data}[[All, 1]]$;

Використовуючи функцію для табличного інтегрування можна упускати написання вбудованого символу InterpolationOrder, тоді порядок інтерполяції за замовчуванням буде k=3.

Як приклад обчислимо значення інтеграла табличної функції y = f(x), значення якої наведені в табл. 1.

Таблиця 1. Табличне представлення функції y = f(x)

x	1	2	3	4	5	6	7	8
y	1	8	27	64	125	216	343	512

У даному випадку крок інтегрування постійний і дорівнює h = 1. З табл. 1 видно, що аналітичний вираз функції має вигляд y = x^3. У лістингу10 наведені процедури обчислення інтеграла в разі задання підінтегральної функції у вигляді таблиці і у вигляді аналітичного виразу y = x^3 .

```
In[260]:= f = {1, 8, 27, 64, 125, 216, 343, 512}
Out[260]= {1, 8, 27, 64, 125, 216, 343, 512}

In[261]:= Integrate[Interpolation[f][x], {x, 1, 8}]
Out[261]= 4095/4

In[262]:= Integrate[x^3, {x, 1, 8}]
Out[262]= 4095/4

In[263]:= f1 = {{1, 1}, {2, 8}, {3, 27}, {4, 64}, {5, 125}, {6, 216},
            {7, 343}, {8, 512}}
Out[263]= {{1, 1}, {2, 8}, {3, 27}, {4, 64},
           {5, 125}, {6, 216}, {7, 343}, {8, 512}}

In[264]:= Integrate[Interpolation[f1][x],
           {x, Min[f1[[All, 1]]], Max[f1[[All, 1]]]}]
Out[264]= 4095/4
```

Лістинг 10

Розв’язання диференційних рівнянь у середовищі Mathematica

Система Mathematica дозволяє вирішувати в аналітичному і чисельному вигляді лінійні і нелінійні диференційні рівняння та системи. Розв’язок диференційних рівнянь та систем здійснюється за допомогою вбудованих функцій.

Аналітичні методи

Аналітичні методи розв'язання диференційних рівнянь в системі Mathematica реалізуються за допомогою двох вбудованих функцій:

$DSolve\ [f,\ y\ [x],\ x];$

$DSolve\ [\{f_1, f_2,\ ...\},\ \{y\ [x_1],\ y\ [x_2],\ ...\},\ \{x_1, x_2,\ ...\}].$

Розглянемо докладно ці функції і наведемо приклади.

Функція $DSolve\ [f,\ y\ [x],\ x]$ призначена для розв’язку диференційного рівняння f щодо функції $y(x)$ з аргументом x. Функція дає загальний розв’язок рівняння з постійними інтегрування, які позначаються $c[i]$.

Диференційне рівняння представляється у довільному вигляді. Наприклад, $y' == 2x\,^\wedge 2 - 1$, $y' - 2x\,^\wedge 2 == -1$ або $y' - 2\ x\,^\wedge 2 + 1 == 0$. Функція дозволяє вирішувати диференційне рівняння будь-якого порядку. Наведемо приклади. Дано наступні диференційні рівняння:

$y' = 2x^2 + 3x - 1,\ y' = 2\ln x + x - 2, y' = 3e^{-2x} - x^2 + x - 1.$

Розв’язок рівнянь наведено у лістингу 11.

```
In[266]:= F = y'[x] == 2 x^2 + 3 x - 1
Out[266]= y'[x] == -1 + 3 x + 2 x^2

In[267]:= DSolve[F, y[x], x]
Out[267]= {{y[x] -> -x + 3 x^2/2 + 2 x^3/3 + C[1]}}

In[268]:= DSolve[y'[x] == 2 Log[x] + x - 2, y[x], x]
Out[268]= {{y[x] -> -4 x + x^2/2 + C[1] + 2 x Log[x]}}

In[269]:= DSolve[y'[x] - e E^(-2 x) + x^2 - x + 1 == 0, y[x], x]
Out[269]= {{y[x] -> -1/2 e E^(-2 x) - x + x^2/2 - x^3/3 + C[1]}}
```

Лістинг 11

З лістингу 11 видно, що при розв’язанні першого рівняння йому присвоєно ім’я F і здійснено введення поза функцією $DSolve$. При вирішенні другого рівняння останнє введено безпосередньо у функцію $DSolve$. При вирішенні третього рівняння воно представлено у вигляді, відмінному від перших двох. З прикладу видно, що програма знайшла спільний розв’язок з довільними постійними інтегрування, а розв’язок зведено до простого інтегрування правої частини рівнянь. Функція $DSolve$ дозволяє також вирішувати диференційні рівняння високого порядку, що демонструється на наступних прикладах:

$y''' = 3x^2 - 2x + 1,\ y''' = y'(x) - 5y(x) + x2 - 1.$

Розв’язок наведено у лістингу 12.

```
DSolve[y'''[x] == 3 x^2 - 2 x + 1, y[x], x]
```

Out[275]= $\left\{\left\{y[x] \to \frac{x^3}{6} - \frac{x^4}{12} + \frac{x^5}{20} + c_1 + x\,c_2 + x^2\,c_3\right\}\right\}$

```
In[276]:= F1 = y'''[x] == y'[x] - 5 y[x] + x^2 - 1
```

Out[276]= $y^{(3)}[x] == -1 + x^2 - 5\,y[x] + y'[x]$

```
In[277]:= DSolve[F1, y[x], x]
```

Out[277]= $\left\{\left\{y(x) \to c_1\, e^{x\,\mathrm{Root}[\#1^3 - \#1 + 5\&, 1, 0]} + c_2\, e^{x\,\mathrm{Root}[\#1^3 - \#1 + 5\&, 2, 0]} + c_3\, e^{x\,\mathrm{Root}[\#1^3 - \#1 + 5\&, 3, 0]} + \frac{1}{125}\left(25\,x^2 + 10\,x - 23\right)\right\}\right\}$

```
In[278]:= N[%]
```

Out[278]= $\{\{y[x] \to 0.008\,(-23. + 10.\,x + 25.\,x^2) + 2.71828^{-1.90416\,x}\,c_1 + 2.71828^{(0.95208 - 1.31125\,i)\,x}\,c_2 + 2.71828^{(0.95208 + 1.31125\,i)\,x}\,c_3\}\}$

Лістинг 12

Розв’язок диференційних рівнянь за умови відомих початкових умов

Для отримання розв’язку диференційних рівнянь за умови відомих початкових умов використовується функція наступна модифікація функції DSolve:

$DSolve\ [f(x, x_0),\ y\ [x],\ x]$,

де $f(x, x_0)$ - диференційне рівняння в сукупності з початковими умовами;

$y(x)$ - шукана функція;

x - незалежна змінна.

Розглянемо приклади окремого розв’язку диференційних рівнянь за допомогою функції $DSolve$:

$y'(x) = 2\ln x + x - 2$ при початковій умові: $y(0) = 1$;

$y''(x) = 3y(x) - e^{2x} + x - 1$ при $y(0) = 1, y'(0) = 0$;

$y''(x) = -5y(x) - 1$ при $y(1) = y'(1) = y''(1) = 0$.

Розв’язок наведено у лістингу 13.

In[279]:= DSolve[{y'[x] == 2 Log[x] + x - 2, y[0] == 1}, y[x], x]

Out[279]= $\left\{\left\{y[x] \to \frac{1}{2}\left(2 - 8x + x^2 + 4x\,\mathrm{Log}[x]\right)\right\}\right\}$

In[280]:= DSolve[{y''[x] == 3 y[x] - E^(2 x) + x - 1, y[0] == 1, y'[0] == 0}, y[x], x]

Out[280]= $\left\{\left\{y[x] \to \left(e^{-\sqrt{3}x}\left(15 - 7\sqrt{3} + 6e^{\sqrt{3}x} + 15e^{2\sqrt{3}x} + 7\sqrt{3}\,e^{2\sqrt{3}x} - 18e^{2x+\sqrt{3}x} - 6e^{\sqrt{3}x}x\right)\right) \Big/ \left(18\left(-2+\sqrt{3}\right)^2\left(2+\sqrt{3}\right)^2\right)\right\}\right\}$

In[281]:= Simplify[%]

Out[281]= $\left\{\left\{y[x] \to \frac{1}{18}e^{-\sqrt{3}x}\left(15 - 7\sqrt{3} + \left(15 + 7\sqrt{3}\right)e^{2\sqrt{3}x} - 18e^{\left(2+\sqrt{3}\right)x} - 6e^{\sqrt{3}x}(-1+x)\right)\right\}\right\}$

In[282]:= DSolve[{y'''[x] == -5 y[x] - 1, y[1] == 0, y'[1] == 0, y''[1] == 0}, y[x], x]

Out[282]= $\left\{\left\{y[x] \to \left(e^{-\frac{5^{1/3}}{2}-5^{1/3}x}\left(e^{\frac{3\,5^{1/3}}{2}}\mathrm{Cos}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2 - 3e^{\frac{5^{1/3}}{2}+5^{1/3}x}\mathrm{Cos}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2 + 2e^{\frac{3}{2}5^{1/3}x}\mathrm{Cos}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]\mathrm{Cos}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}x\right] + e^{\frac{3\,5^{1/3}}{2}}\mathrm{Sin}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2 - 3e^{\frac{5^{1/3}}{2}+5^{1/3}x}\mathrm{Sin}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2 + 2e^{\frac{3}{2}5^{1/3}x}\mathrm{Sin}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]\mathrm{Sin}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}x\right]\right)\right) \Big/ \left(15\left(\mathrm{Cos}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2 + \mathrm{Sin}\left[\frac{1}{2}\sqrt{3}\,5^{1/3}\right]^2\right)\right)\right\}\right\}$

Лістинг 13

Розв'язок систем диференційних рівнянь в аналітичному вигляді

Системи диференційних рівнянь в аналітичному вигляді вирішуються також за допомогою вбудованої функції *DSolve* , яка в даному випадку має вигляд:

$DSolve\,[\{f_1, f_2, ...\}, \{y_1(x), y_2(x), ...\}, x]$

де f_i - і-те рівняння системи;

$y_i(x)$ - і-те шукане невідоме;

x - незалежна змінна.

Приклади використання функції *DSolve* для випадку розв'язання систем диференційних рівнянь наведено нижче:

$$\begin{cases} x'(t) = y(t) + z(t) \\ y'(t) = x(t) + 3z(t) \\ z'(t) = x(t) + y(t). \end{cases}$$

Розв'язок наведено у лістингу 14.

```
In[283]:= f = {x'[t] == y[t] + z[t], y'[t] == x[t] + z[t], z'[t] == x[t] + y[t]}
Out[283]= {x′[t] == y[t] + z[t], y′[t] == x[t] + z[t], z′[t] == x[t] + y[t]}

In[284]:= DSolve[f, {x[t], y[t], z[t]}, t]
```

$$\text{Out[284]=}\ \left\{\left\{x[t] \to \frac{1}{3}e^{-t}\left(2+e^{3t}\right)c_1 + \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_2 + \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_3,\right.\right.$$
$$y[t] \to \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_1 + \frac{1}{3}e^{-t}\left(2+e^{3t}\right)c_2 + \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_3,$$
$$\left.\left. z[t] \to \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_1 + \frac{1}{3}e^{-t}\left(-1+e^{3t}\right)c_2 + \frac{1}{3}e^{-t}\left(2+e^{3t}\right)c_3\right\}\right\}$$

Лістинг 14

З лістингу 14 видно, що розв'язок отримано в загальному вигляді, так як початкові умови не були задані.

За умови підстановки початкових умов: $x(0)=1, y(0)=z(0)=0$ запис функції *DSolve* є наступним:

$$DSolve[\{x'[t] == y[t]+z[t], y'[t] == x[t]+z[t], z'[t] = x[t]+y[t],$$
$$x[0]=1, y[0] == z[0] == 0\}, \{x[t], y[t], z[t]\}, t].$$

Чисельні методи розв'язку диференційних рівнянь

Чисельні методи розв'язку диференційних рівнянь у системі Mathematica реалізуються за допомогою наступних двох вбудованих функцій:

$$NDSolve\ [f, y[x], \{x, x_{\min}, x_{\max}\}]$$
$$NDSolve\ [\{f_1, f_2, ..., y_1(x_0), y_2(x_0), ...\}, \{y_1[x], y_2[x], ...\}, \{x, x_{\min}, x_{\max}\}]$$

де f - диференційне рівняння і початкові умови;

f_i - і-те рівняння системи диференційних рівнянь;

$y[x]$ - шукана функція;

$y_i[x]$- і-та шукана функція системи диференційних рівнянь;

$y_i(x_0)$ - і-та початкова умова;

$x_{\min}, x_{\max}$ - мінімальне і максимальне значення незалежної змінної;

x - аргумент шуканої функції.

Функції чисельного розв'язку диференційних рівнянь і систем мають опцію *StartingStepSize*, яка визначає величину початкового кроку інтегрування.

Чисельні методи найбільш часто застосовуються в тих випадках, коли рівняння в аналітичному вигляді системою не вирішується або зовсім не має аналітичного розв'язку. Такими є більшість нелінійних рівнянь. Далі детально викладаються вбудовані функції, способи їх реалізації та наводяться приклади.

Функція $NDSolve\ [f, y[x], \{x, x_{\min}, x_{\max}\}]$

Ця функція вирішує диференційне рівняння *n*-го порядку, обчислюючи шукану функцію *y(x)* в діапазоні незалежної змінної *x* від $x_{\min}$ до $x_{\max}$. Розв'язок можна отримати у вигляді таблиці або графіка.

Методику розв'язання задачі показано на прикладі такого диференційного рівняння:

$$y'(t) - xy(x) = 1,$$

при початкових умовах $y(0) = 1$. Розв'язок слід отримати в табличному і графічному вигляді в діапазоні від 0 до 5 з кроком 0.5.

Методика вирішення диференційного рівняння з допомогою функції *NDSolve* складається з виконання наступних операцій:

1. Введення функції *NDSolve*, яка в нашому прикладі має вигляд:

$$NDSolve[\{y'[x] == xy[x] + 1, y[0] == 1\}, y[x], \{x, 0, 5\}].$$

2. Отримання розв'язку шляхом одночасного натискання клавіш <Shilit> + <Enter>. Розв'язок буде отримано у вигляді повідомлення без виведення самого розв'язку на екран.

3. Введення функції *Table* для отримання розв'язку в табличній формі. У нашому прикладі ця функція матиме вигляд:

$$Table[\{x, y[x] /.Out[292]\}, \{x, 0, 5, 0.5\}].$$

Тут *Out [292]* - це 292 рядок, в якому знаходиться розв'язок рівняння. Результатом є вектор, представлений у вигляді рядка.

4. Введення функції *TableForm [%]* для отримання розв'язку у формі таблиці. Результатом є функція *y(x)*, представлена у вигляді таблиці.

5. Введення функції $Plot[\{y[x] /.Out[292]\}, \{x, 0, 5\}]$. Результатом є графік функції.

Розв'язок завдання наведено у лістингу 15. Спочатку наведено розв'язання рівняння в аналітичному вигляді.

У наступному прикладі необхідно вирішити рівняння вищого порядку:

$$y'''(x) - 3x^2 y''(x) - 2xy'(x) - y(x) = 1,$$

при початкових умовах $y(1) = 1, y'(1) = y''(1) = 0$. Розв'язок отримати в діапазоні *x*від 0 до 3 з кроком 0.5 при поданні рішення в табличному вигляді та в діапазоні від 0 до 2 - у графічному. Розв'язок наведено у лістингу 16.

In[285]:=
```
DSolve[{y'[x] == x y[x] + 1, y[0] == 1}, y[x], x]
```

Out[285]= $\left\{\left\{y[x] \to \frac{1}{2} e^{\frac{x^2}{2}} \left(2 + \sqrt{2\pi}\, \mathrm{Erf}\left[\frac{x}{\sqrt{2}}\right]\right)\right\}\right\}$

In[292]:=
```
NDSolve[{y'[x] == x y[x] + 1, y[0] == 1}, y[x], {x, 0, 5}]
{{y[x] → InterpolatingFunction[{{0., 5.}}, <>][x]}}
```

Out[292]= {{y[x] → InterpolatingFunction[Domain: {{0., 5.}} Output: scalar][x]}}

In[293]:=
```
Table[{x, y[x] /. Out[292]}, {x, 0, 5, 0.5}]
```

Out[293]= {{0., {1.}}, {0.5, {1.67697}}, {1., {3.05941}}, {1.5, {6.42488}}, {2., {16.2285}}, {2.5, {50.9309}}, {3., {202.532}}, {3.5, {1029.82}}, {4., {6716.79}}, {4.5, {56 240.7}}, {5., {604 648.}}}

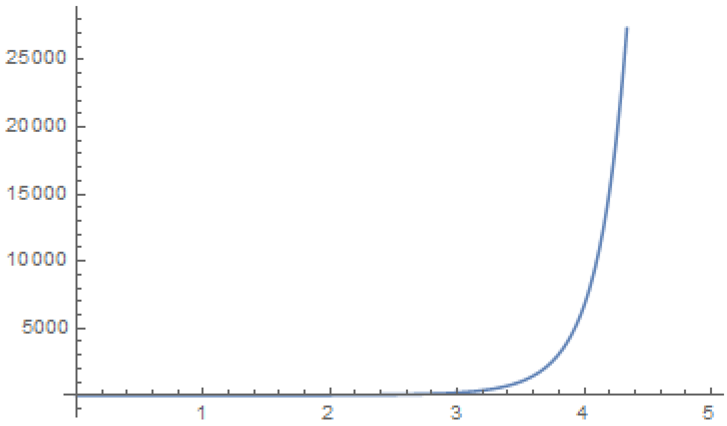
In[294]:=
```
TableForm[%]
```

Out[294]//TableForm=

0.	1.
0.5	1.67697
1.	3.05941
1.5	6.42488
2.	16.2285
2.5	50.9309
3.	202.532
3.5	1029.82
4.	6716.79
4.5	56 240.7
5.	604 648.

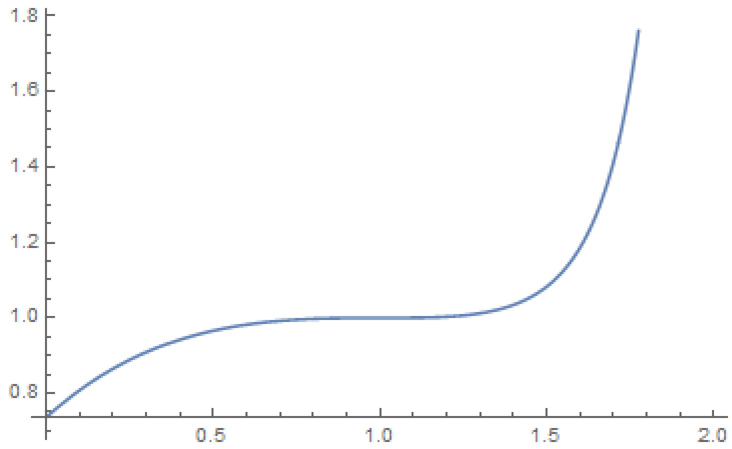
In[295]:=
```
Plot[y[x] /. Out[292], {x, 0, 5}]
```

Out[295]=



Лістинг 15

```
In[312]:= NDSolve[{y'''[x] == 3 x^(2) y''[x] + 2 x y'[x] + y[x] + 1,
              y[1] == 1, y'[1] == 0, y''[1] == 0}, y[x], {x, 0, 5}]
           {{y[x] → InterpolatingFunction[{{0., 5.}}, <>][x]}}

Out[312]= {{y[x] → InterpolatingFunction[ Domain: {{0., 5.}}
                                           Output: scalar ][x]}}

In[314]:= Table[{x, y[x] /. Out[312]}, {x, 0, 3, 0.5}]

Out[314]= {{0., {0.736358}}, {0.5, {0.966263}}, {1., {1.}}, {1.5, {1.08349}},
           {2., {6.94346}}, {2.5, {5233.08}}, {3., {2.36809 × 10^8}}}

In[315]:= TableForm[%]

Out[315]//TableForm=
           0.      0.736358
           0.5     0.966263
           1.      1.
           1.5     1.08349
           2.      6.94346
           2.5     5233.08
           3.      2.36809 × 10^8

In[316]:= Plot[{y[x] /. Out[312]}, {x, 0, 2}]

Out[316]=
```



Лістинг 16

Функція NDSolve $[\{f_1, f_2, ..., y_1(x_0), y_2(x_0), ...\}, \{y_1[x], y_2[x], ...\}, \{x, x_{\min}, x_{\max}\}]$

Ця функція вирішує систему диференційних рівнянь *n*-го порядку, обчислюючи шукані функції $y_1[x], y_2[x], ...$ в діапазоні незалежної змінної *x* від $x_{\min}$ до $x_{\max}$. Розв'язок можна отримати у вигляді таблиці або графіків.

Методику використання функції показано на такому прикладі:

$$\begin{cases} p_0'(t) = -0.9\,p_0(t) + 2\,p_1(t) \\ p_1'(t) = 0.9\,p_0(t) - 2.9\,p_1(t) + 4\,p_2(t) \\ p_2'(t) = 0.9\,p_1(t) - 4\,p_2(t), \end{cases}$$

за наступних початкових умов: $p_0(0) = 1, p_1(0) = p_2(0) = 0$. Розв'язок отримати у вигляді таблиць і графіків.

У цьому випадку функція *NDSolve* буде мати вигляд:

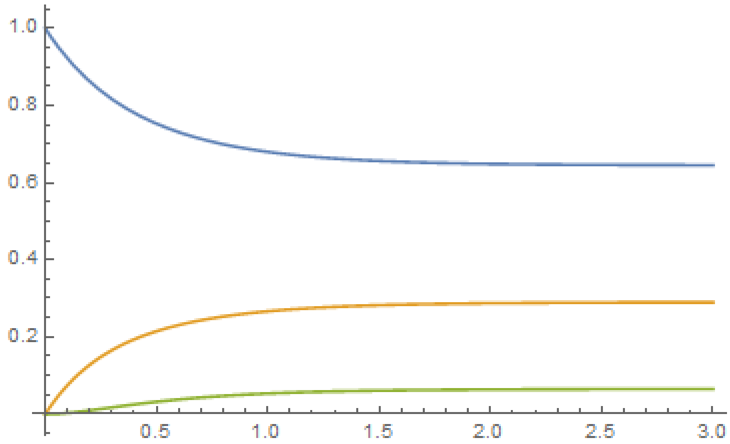
NDSolve $[\{p_0'[t]=-0.9\,p_0[t]+2p_1[t],\ p_1'[t]=0.9\,p_0[t]-2.9p_1[t]+4p_2[t],\ p_2'[t]=0.9p_1[t]-4p_2[t],\ p_0[0]=1, p_1[0]=p_2[0]=0\},\ \{p_0[t],\ p_1[t],\ p_2[t]\},\ \{t,\ 0,\ 100\}]$.

Розв'язок наведено у лістингу 17. Таблиця представлена в діапазоні t від 0 до 1 з кроком 0.1, а графік - в діапазоні t від 0 до 3.

```
In[317]:= NDSolve[{p0'[t] == -0.9 p0[t] + 2 p1[t],
            p1'[t] == 0.9 p0[t] - 2.9 p1[t] + 4 p2[t], p2'[t] == 0.9 p1[t] - 4 p2[t],
            p0[0] == 1, p1[0] == p2[0] == 0}, {p0[t], p1[t], p2[t]}, {t, 0, 100}]
          {{p0[t] → InterpolatingFunction[{{0., 100.}}, <>][t],
            p1[t] → InterpolatingFunction[{{0., 100.}}, <>][t],
            p2[t] → InterpolatingFunction[{{0., 100.}}, <>][t]}}

Out[317]= {{p0[t] → InterpolatingFunction[ Domain: {{0., 100.}} Output: scalar ][t],
            p1[t] → InterpolatingFunction[ Domain: {{0., 100.}} Output: scalar ][t],
            p2[t] → InterpolatingFunction[ Domain: {{0., 100.}} Output: scalar ][t]}}

In[318]:= Table[{t, {p0[t], p1[t], p2[t]} /. Out[317]}, {t, 0, 1, 0.1}]

Out[318]= {{0., {{1., 0., 0.}}}, {0.1, {{0.921667, 0.0751894, 0.00314313}}},
           {0.2, {{0.862132, 0.127986, 0.0098828}}},
           {0.3, {{0.81629, 0.166013, 0.0176963}}},
           {0.4, {{0.780636, 0.194028, 0.0253356}}},
           {0.5, {{0.752693, 0.215063, 0.0322435}}},
           {0.6, {{0.730668, 0.231105, 0.038227}}},
           {0.7, {{0.713235, 0.243489, 0.0432764}}},
           {0.8, {{0.699393, 0.253141, 0.0474669}}},
           {0.9, {{0.688377, 0.260716, 0.0509063}}},
           {1., {{0.679598, 0.266694, 0.053708}}}}
```

Лістинг 17 (Частина 1 з 2)

```
In[319]:= TableForm[%]

Out[319]//TableForm=
0.     1. 0. 0.
0.1    0.921667 0.0751894 0.00314313
0.2    0.862132 0.127986 0.0098828
0.3    0.81629 0.166013 0.0176963
0.4    0.780636 0.194028 0.0253356
0.5    0.752693 0.215063 0.0322435
0.6    0.730668 0.231105 0.038227
0.7    0.713235 0.243489 0.0432764
0.8    0.699393 0.253141 0.0474669
0.9    0.688377 0.260716 0.0509063
1.     0.679598 0.266694 0.053708

In[325]:= Plot[{{p0[t] /. Out[317]}, {p1[t] /. Out[317]},
   {p2[t] /. Out[317]}}, {t, 0, 3}]

Out[325]=
```



Лістинг 17 (Частина 2 з 2)

Система Mathematica, як і будь-яка інша система комп'ютерної алгебри, не є ідеальною щодо розв'язання диференційних рівнянь. Одержаний розв'язок рідко коли збігається з відповіддю, наявною в математичних довідниках. Не рідкі випадки, коли вбудована функція не дає розв'язку або він є помилковим, хоча рівняння досить просте. Наведемо такі приклади. Нехай потрібно вирішити наступні рівняння і системи рівнянь:

$y''(x) = -y(x) + tgx;$

$y''(x) = -y(x) + 4x\sin x;$

$x'(t) + y'(t) - tx(t) = t;$

$x'(t) + y'(t) + y(t) = t(t+2);$

$3z(x)z'(x) - 2x = 0.$

Систему рівнянь вирішити аналітичним і чисельним методами при початкових умовах, $x(0)=1, y(0)=0$ в діапазоні x від -1 до 1.

Розв'язки рівнянь наведено у лістингу 18.

```
In[326]:= DSolve[y''[x] + y[x] == Tan[x], y[x], x]

Out[326]= {{y[x] → c1 Cos[x] + Cos[x] Log[Cos[x/2] - Sin[x/2]] -
      Cos[x] Log[Cos[x/2] + Sin[x/2]] + c2 Sin[x]}}

In[327]:= DSolve[y''[x] + y[x] == 4 x Sin[x], y[x], x]

Out[327]= {{y[x] → c1 Cos[x] + c2 Sin[x] + 1/2 (-2 x^2 Cos[x] + Cos[x] Cos[2 x] -
        2 x Cos[2 x] Sin[x] + 2 x Cos[x] Sin[2 x] + Sin[x] Sin[2 x])}}

In[328]:= Simplify[%]

Out[328]= {{y[x] → (1/2 - x^2 + c1) Cos[x] + (x + c2) Sin[x]}}

In[329]:= DSolve[{x'[t] + y'[t] - t x[t] == t, x'[t] + y'[t] + y[t] == t (t + 2),
      x[0] == 1, y[0] == 0}, {x[t], y[t]}, t]

Out[329]= DSolve[{-t x[t] + x'[t] + y'[t] == t,
      y[t] + x'[t] + y'[t] == t (2 + t), x[0] == 1, y[0] == 0}, {x[t], y[t]}, t]

In[330]:= NDSolve[{x'[t] + y'[t] - t x[t] == t, x'[t] + y'[t] + y[t] == t (t + 2),
      x[0] == 1, y[0] == 0}, {x[t], y[t]}, {t, -1, 1}]

Out[330]= {{x[t] → InterpolatingFunction[ Domain: {{-1., 1.}} Output: scalar ][t],
      y[t] → InterpolatingFunction[ Domain: {{-1., 1.}} Output: scalar ][t]}}

In[331]:= DSolve[3 z[x]^2 × z'[x] - 2 x == 0, z[x], x]

Out[331]= {{z[x] → (x^2 + 3 c1)^(1/3)},
      {z[x] → -(-1)^(1/3) (x^2 + 3 c1)^(1/3)}, {z[x] → (-1)^(2/3) (x^2 + 3 c1)^(1/3)}}
```

Лістинг 18

На основі лістингу можна дати наступні коментарі. Розв’язок першого і другого рівнянь є вірними, але не збігаються з довідковими даними, в яких наводяться такі відповіді:

$$y(x) = c_1 \cos x + c_2 \sin x - \cos x \ln(\mathrm{tg}(\pi/4 + x/2)),$$

$$y(x) = c_1 \cos x + c_2 \sin x + x(\sin x - x \cos x).$$

Отримані рішення можна істотно спростити за допомогою функції *Simplify [%]*.

Розв’язок системи рівнянь не отриманий ні аналітичним, ні чисельним методами, хоча такий розв’язок існує.

Цікавою є відповідь розв’язку третього рівняння-три рівноцінних результати.

При вирішенні диференційних рівнянь чисельними методами можуть виникати неприпустимо великі помилки за рахунок методичних помилок і помилок вибору кроку інтегрування. Завжди необхідно пам'ятати, що при комп'ютерних технологіях розв’язку диференційних рівнянь необхідна перевірка достовірності отриманих результатів.

Хід роботи

1. Обчислити суму ряду з кроком Δn = 1 згідно з варіантом, вказаним у табл. 1.

Таблиця 1. Сума ряду

Вар.	Ряд	Вар.	Ряд	Вар.	Ряд	Вар.	Ряд
1,11	$\sum_{k=1}^{n} 2k-1$	4,14	$\sum_{k=0}^{\infty} aq^k$	7,17	$\sum_{k=1}^{\infty} \frac{1}{k2^k}$	10,20	$\sum_{k=1}^{n-1} sin\left(\frac{\pi k}{n}\right)$
2,12	$\sum_{k=1}^{n} k!k$	5,15	$\sum_{k=1}^{\infty} \frac{1}{(2k-1)^2}$	8,18	$\sum_{k=1}^{\infty} \frac{x^k}{k}$		
3,13	$\sum_{k=1}^{n} \frac{k^2+k-1}{(k+2)!}$	6,16	$\sum_{k=1}^{\infty} \frac{1}{4k^2-1}$	9,19	$\sum_{k=1}^{\infty} \frac{1}{x^k k}$		

2. Обчислити добуток ряду з кроком Δn = 1 згідно з варіантом, вказаним у табл. 2.

Таблиця 2. Добуток ряду

Вар.	Ряд	Вар.	Ряд	Вар.	Ряд	Вар.	Ряд
1,11	$\prod_{k=2}^{\infty}\left(1-\frac{1}{(2k-1)^2}\right)$	4,14	$\prod_{k=2}^{\infty}\left(1-\frac{1}{k^2}\right)$	7,17	$x\prod_{k=1}^{\infty}\left(1+\frac{x^2}{(k\pi)^2}\right)$	10,20	$x\prod_{k=1}^{\infty}\left(1-\frac{x^2}{(k\pi)^2}\right)$
2,12	$\prod_{k=0}^{\infty}\left(1+x^{2^k}\right)$	5,15	$\prod_{k=1}^{\infty}\left(1-\frac{1}{(2k)^2}\right)$	8,18	$2^{n-1}\prod_{k=0}^{n-1} sin\left(x+\frac{k\pi}{n}\right)$		
3,13	$\prod_{k=1}^{\infty}\left(1+\frac{(-1)^{k+1}}{2k-1}\right)$	6,16	$\prod_{k=1}^{\infty}\left(1-\frac{4x^2}{\pi^2(2k+1)^2}\right)$	9,19	$x\prod_{k=1}^{\infty} cos\left(\frac{x}{2^k}\right)$		

3. Обчислити границю функції згідно з варіантом, вказаним у табл.3.

Таблиця 3. Границя функції

Вар.	Границя	Вар.	Границя	Вар.	Границя	Вар.	Границя
1,11	$\lim_{x\to\infty}(1+x)^{\frac{1}{x}}$	4,14	$\lim_{x\to 1}(1-x)^{\frac{1}{1-x}}$	7,17	$\lim_{x\to 0}\left(\frac{1-a^x}{ax}\right)$	10,20	$\lim_{x\to 2}\left(\frac{n^2-n^x}{x-2}\right)$
2,12	$\lim_{x\to 0}\left(\frac{a^x+b^x}{2}\right)^{\frac{1}{x}}$	5,15	$\lim_{x\to\infty}(1-x)^{\frac{1}{1-x}}$	8,18	$\lim_{x\to\infty}\left(2a\frac{1-e^{-ax}}{2-e^{-ax}}\right)$		
3,13	$\lim_{x\to\infty}\left(1+\frac{1}{x}\right)^{\frac{1}{x}}$	6,16	$\lim_{x\to 0}\left(\frac{1-e^{ax}}{ln(1-x)}\right)$	9,19	$\lim_{x\to a}\left(\frac{x-a}{ln(x-a+1)}\right)$		

4. Розкласти функцію у степеневий ряд і обчислити її значення в точці x_0 згідно з варіантом, вказаним у табл. 4.

Таблиця 4. Степеневий ряд функції

Вар.	Функція	Кількість членів	Точка розкладу	Вар.	Функція	Кількість членів	Точка розкладу
1,11	$\mathbf{arcsin}(5x)$	5	$x_0 = 0.3$	6,16	$x/(1-x^2)$	7	$x_0=0.5$
2,12	$\mathbf{ln}(2x+1)$	7	$x_0 = 4$	7,17	$\mathbf{ln}((1+x)/(1-x))$	7	$x_0=0.9$
3,13	$\mathbf{sin}(3x^2)$	3	$x_0 = \pi/3$	8,18	$(1+x)^m$	8	$x_0=3$, m=5
4,14	$\mathbf{sin}(x)/x$	5	$x_0 = \pi/4$	9,19	$\mathbf{ln}\left(x+\sqrt{x^2+1}\right)$	8	$x_0=0.8$
5,15	$1-e^x/x$	5	$x_0 = 10$	10,20	$tg(x)$	6	$x_0=7\pi/8$

5. Обчислити похідну функції в точці x, згідно з варіантом, вказаним у табл. 5.

Таблиця 5. Аналітичний вираз функції

Вар.	Функція	x_1-x_2	Вар.	Рівняння	x	Вар.	Рівняння	x
1,11	$(\mathbf{sin}\,ax+\mathbf{cos}\,ax)^2$	0..2.5	4,14	$(x-1)/(x+1)$	-9..-6	7,17	$x\,\mathbf{cos}\,x$	0..π/4
2,12	ax^3+bx-c	1.5..3.4	5,15	$\sqrt{a+bx^2}$	2.3-12.2	8,18	$\mathbf{sin}(x)^2\,\mathbf{cos}(x)^4$	0..π/12
3,13	xe^{-ax}	-3..-1	6,16	$\mathbf{sin}(ax)/(x+b)$	8.4-10	9,19	$x^2\,\mathbf{arcsin}(x)$	0..π/6
						10,20	$x^2\,\mathbf{ln}\,x$	0..e

6. Обчислити інтеграл функції у аналітичному виді і знайти його значення у заданих межах x_1-x_2 інтегрування згідно з варіантом, вказаним у табл. 5.

7. Обчислити значення інтегралу чисельним методом згідно з варіантом, вказаним у табл. 5.

8. Розв'язати диференційне рівняння аналітичним і чисельним методом згідно з варіантом, вказаним у табл. 6.

Таблиця 6. Диференційне рівняння

Вар.	Функція	Вар.	Рівняння	Вар.	Рівняння
1,11	$y''-0.5y'-0.5y=e^{0.5x}$ $y(0)=1, y'(1)=0$	4,14	$(1+x)y''+y'=0$ $y(0)=-2, y'(0)=0$	7,17	$8y''+4y'+4y=0$ $y(0)=1, y'(0)=3$
2,12	$y''-3y'+2y=x^2+3x$ $y(0)=-1, y'(0)=3$	5,15	$y'=0.5xy$ $y(0)=1$	8,18	$y''-3y'=x^2+3x$ $y(0)=1, y'(0)=3$
3,13	$y'=-2y/(100+x)$ $y(0)=0$	6,16	$8y''+2y'-3y=0$ $y(0)=-1, y'(0)=1$	9,19	$y''-2y'+y=xe^x$ $y(0)=1, y'(0)=1$
				10,20	$y'=-xy/(1+x^2)$

					$y(0) = 0$

9. Розв'язати диференційне рівняння аналітичним методом за умови відомих початкових умов згідно з варіантом, вказаним у табл. 7.

Таблиця 7. Диференційне рівняння

Вар.	Функція	Вар.	Рівняння	Вар.	Рівняння
1,11	$y' - \frac{4}{x}y = x\sqrt{y}, y(1) = 1$	4,14	$y'' + y = 4xe^x,$ $y(0) = -2, y'(0) = 0$	7,17	$y'' + 9y = 6cos(3x),$ $y(0) = 1, y'(0) = 3$
2,12	$y'' + 3y = -2y,$ $y(0) = -1, y'(0) = 3$	5,15	$y'' + y = sin(x),$ $y(0) = 1, y'(0) = 2$	8,18	$y'' - y = e^x,$ $y(0) = 0, y'(0) = -1$
3,13	$y''' = -8y,$ $y(0) = 0, y'(0) = 1, y''(0) = 1$	6,16	$y'' + 4y' + 4y = 8xe^{2x},$ $y(0) = -1, y'(0) = 1$	9,19	$y'' + 2y' - 3y = x^2e^x,$ $y(0) = 1, y'(0) = 1$
				10,20	$y'' - 4y' + 5y = 0,$ $y(0) = 0, y'(0) = 1$

10. Розв'язати систему диференційних рівнянь аналітичним методом згідно з варіантом, вказаним у табл. 8.

Таблиця 8. Диференційне рівняння

Вар.	Функція	Вар.	Рівняння	Вар.	Рівняння
1,11	$\begin{cases} x' = y + 2e^t; \\ y' = x + t^2. \end{cases}$	4,14	$\begin{cases} x' = 2x - 4y + 4e^{-2t}; \\ y' = 2x - 2y. \end{cases}$	7,17	$\begin{cases} x' = 5x - 3y + 2e^{3t}; \\ y' = x + y + 5e^{-t}. \end{cases}$
2,12	$\begin{cases} x' = y - 5cos(t); \\ y' = 2x + y. \end{cases}$	5,15	$\begin{cases} x' = 2x - 4y + 4e^{-2t}; \\ y' = 2x - 2y. \end{cases}$	8,18	$\begin{cases} x' = 2x + y + e^t; \\ y' = -2x + 2t. \end{cases}$
3,13	$\begin{cases} x' = 3x + 2y + 4e^{5t}; \\ y' = x + 2y. \end{cases}$	6,16	$\begin{cases} x' = 2y - x + 1; \\ y' = 3y - 2x. \end{cases}$	9,19	$\begin{cases} x' = x + 2y; \\ y' = x - 5sin(t). \end{cases}$
				10,20	$\begin{cases} x' = 2x - 4y; \\ y' = x - 3y + 3e^t. \end{cases}$

Контрольні запитання

1. Вкажіть назву функції, призначену для розрахунку суми рядів і перелічіть її параметри.
2. Опишіть призначення і параметри функції Limit[].
3. Перелічіть параметри функції Series[].
4. Перелічіть функції інтегрування і диференціювання функцій.
5. Опишіть процедуру розрахунку інтегралів функцій заданих таблично.
6. Перелічіть модифікації функцій, призначені для розв'язку диференційних рівнянь.
7. Перелічіть модифікації функцій, призначені для розв'язку систем диференційних рівнянь.
8. Опишіть особливості вирішення диференційних рівнянь з відомими і невідомими початковими умовами.
9. Опишіть особливості вирішення диференційних рівнянь чисельними і аналітичними методами.

10. Опишіть методику вирішення диференційного рівняння з допомогою функції *NDSolve*.

Лабораторна № 6
Комп’ютерні технології інтерполяції в середовищі Mathematica

Мета: вивчення функцій інтерполяції програми Mathematica.

Короткі теоретичні відомості

Інтерполяція, точна у вузлах

В середовищі Mathematica інтерполяція, точна у вузлах, може бути реалізована наступними методами:

- універсальним;
- за допомогою універсальних функцій InterpolatingPolynomial і Interpolation.

Універсальний метод

Універсальний метод потребує вирішення систем алгебраїчних рівнянь, які були отримані на основі даних функції $y = f(x)$, яка представлена у вигляді таблиці чи матриці.

Приклади інтерполяції універсальним методом наведені нижче.

Функція $y = f(x)$ задана у вигляді табл. 1.

Таблиця 1. Функція $y = f(x)$ в табличній формі

x	1	2	3	4
y	6,2	4,1	1,9	0,6

Необхідно розв’язати задачу інтерполяції, яка є точною у вузлах, якщо функція $y = \varphi(x)$ є поліномом. Так як кількість вузлів $n = 4$, то степінь полінома має бути не вищим за $n - 1$. Тобто $y = a_0 + a_1x + a_2x^2 + a_3x^3$.

Складемо систему рівнянь:

$$a_0 + a_1 \cdot 1 + a_2 \cdot 1^2 + a_3 \cdot 1^3 = 6{,}2$$
$$a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 + a_3 \cdot 2^3 = 4{,}1$$
$$a_0 + a_1 \cdot 3 + a_2 \cdot 3^2 + a_3 \cdot 3^3 = 1{,}9$$
$$a_0 + a_1 \cdot 4 + a_2 \cdot 4^2 + a_3 \cdot 4^3 = 0{,}6$$

Розв’язок отримаємо за допомогою функції $Solve\big(F, (a_0, a_1, a_2, a_3)\big)$, де $F -$ вихідна система рівнянь (див. лістинг 1).

```
In[1]:= F = {a0 + a1 + a2 + a3 == 6.2, a0 + 2 a1 + 2^2 a2 + 2^3 a3 == 4.1,
          a0 + 3 a1 + 3^2 a2 + 3^3 a3 == 1.9, a0 + 4 a1 + 4^2 a2 + 4^3 a3 == 0.6}
        Solve[F, {a0, a1, a2, a3}]

Out[1]= {a0 + a1 + a2 + a3 == 6.2, a0 + 2 a1 + 4 a2 + 8 a3 == 4.1,
         a0 + 3 a1 + 9 a2 + 27 a3 == 1.9, a0 + 4 a1 + 16 a2 + 64 a3 == 0.6}

Out[2]= {{a0 → 7.2, a1 → -0.116667, a2 → -1.05, a3 → 0.166667}}
```

Лістинг 1

У результаті отриманого розв’язку інтерполяційна формула буде мати вигляд:

$$y = 7{,}2 - 0.116667x - 1{,}05x^2 + 0{,}166667x^3.$$

Задачу інтерполяції, точної у вузлах також можна вирішити за допомогою матричного методу розв'язку системи лінійних рівнянь. Перевірка правильності розв'язку задачі виконують табуляцією отриманої формули й порівнянням результатів з вихідними даними, наприклад за допомогою функції $Table[f(x),\{x,x_H,x_K,h\}]$, де $f(x)-$ функція, що підлягає табуляції, $x-$аргумент функції $f(x)$, x_H,x_K- початкове й кінцеве значення аргументу, $h-$ крок таблиці, якщо $h=1$, то ним можна знехтувати.

У попередньому прикладі число рівнянь і число невідомих однакове. При розв'язуванні практичних задач майже завжди кількість значень табульованої функції перевищує степінь алгебраїчних рівнянь. У таких випадках обмежене число вузлів інтерполяції доводиться обирати з усього діапазону вихідних даних. Проілюструємо це на прикладі табульованої функції, навденої у табл. 2.

Таблиця 2. Табульована функція

x	3	6	9	12	15	18	21	24	27	30
y	26	90	180	300	500	700	1000	1200	1500	2000

Нехай функція інтерполяції є поліномом другого степеня $y=a_0+a_1x+a_2x^2$. Розрахунок коефіцієнтів інтерполяційного полінома наведено у лістингу 2. В результаті порівняння результатів інтерополяції з вихідними даними можна зробити висновок, про наявність похибки інтерполяції при деяких значеннях аргументу x. Для детального аналізу величини похибки на рис. 1 побудовано функції інтерполяції і даних.

```
In[3]:= F = {a0 + 6 a1 + 6^2 a2 == 90, a0 + 18 a1 + 18^2 a2 == 700,
           a0 + 27 a1 + 27^2 a2 == 1500}
        Solve[F, {a0, a1, a2}]

Out[3]= {a0 + 6 a1 + 36 a2 == 90,
         a0 + 18 a1 + 324 a2 == 700, a0 + 27 a1 + 729 a2 == 1500}
```

Out[4]= $\left\{\left\{a0 \to -\frac{135}{7},\ a1 \to \frac{925}{126},\ a2 \to \frac{685}{378}\right\}\right\}$

```
In[5]:= N[%]

Out[5]= {{a0 -> -19.2857, a1 -> 7.34127, a2 -> 1.81217}}

In[6]:= y = -135 / 7 + 925 / 126 x + 685 / 378 x^2
        Table[y, {x, 3, 30, 3}]
```

Out[6]= $-\frac{135}{7}+\frac{925\,x}{126}+\frac{685\,x^2}{378}$

Out[7]= $\left\{\frac{400}{21},\ 90,\ \frac{1355}{7},\ \frac{6925}{21},\ \frac{3490}{7},\ 700,\ \frac{19615}{21},\ \frac{8405}{7},\ 1500,\ \frac{38470}{21}\right\}$

```
In[8]:= N[%]

Out[8]= {19.0476, 90., 193.571, 329.762,
         498.571, 700., 934.048, 1200.71, 1500., 1831.9}
```

Лістинг 2

```
In[13]:= f1 = {{3, 26}, {6, 90}, {9, 180}, {12, 300}, {15, 500},
            {18, 700}, {21, 1000}, {24, 1200}, {27, 1500}, {30, 2000}};
         r1 = ListPlot[f1, AxesLabel → {"x", "y"},
            PlotStyle → PointSize[0.02]];
         r2 = Plot[-135 / 7 + 925 / 126 x + 685 / 378 x^2, {x, 2, 40}];
         Show[{r1, r2}]
```

Out[16]=

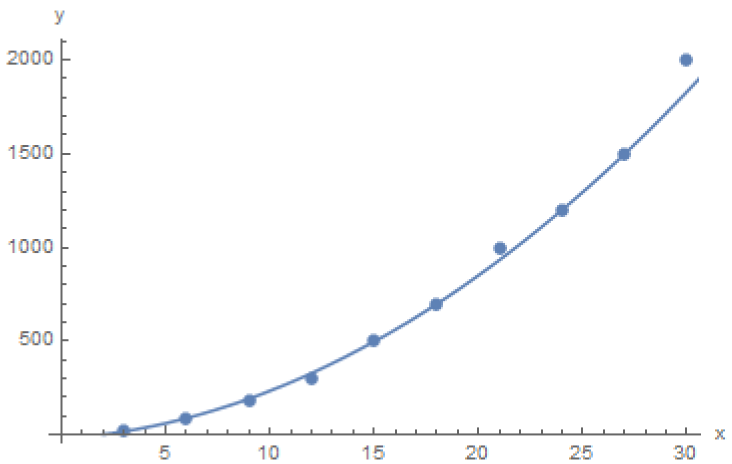


Рис. 1. Графіки вихідної функції та табуляції

Обчислимо абсолютну ε і відносну δ середньоквадратичну похибку інтерполяції за такими формулами:

$$\varepsilon = \sqrt{\frac{\sum_{i=1}^{n} \Delta_i^2}{n}}, \delta = \frac{\varepsilon}{y_{\min}} \cdot 100\%, \qquad (1)$$

де $\Delta_i = y(x_i) - \varphi(x_i)$ – різниця між значеннями табульованої функції $y(x_i)$ і функції інтерполяції $\varphi(x_i)$,

n – кількість значень табульованої функції,

y_{min} – мінімальне значення функції y(x).

Розрахунок значень абсолютної і відносної і похибок наведено у лістингу 3.

```
In[1]:= v1 = {26, 90, 180, 300, 500, 700, 1000, 1200, 1500, 2000}
        v2 = {26.0476, 90., 180.571, 300.762, 500.553,
          700., 1000.048, 1200.71, 1500., 2000.9}
        z = v1 - v2

Out[1]= {26, 90, 180, 300, 500, 700, 1000, 1200, 1500, 2000}

Out[2]= {26.0476, 90., 180.571, 300.762, 500.553,
         700., 1000.05, 1200.71, 1500., 2000.9}

Out[3]= {-0.0476, 0., -0.571, -0.762,
         -0.553, 0., -0.048, -0.71, 0., -0.9}

In[5]:= z^2

Out[5]= {0.00226576, 0., 0.326041, 0.580644,
         0.305809, 0., 0.002304, 0.5041, 0., 0.81}

In[6]:= Plus[0.0022657599999999236`, 0.`,
        сложить
         0.32604099999999764`, 0.5806440000000007`,
         0.305808999999997`, 0.`, 0.002304000000000175`,
         0.5041000000000516`, 0.`, 0.8100000000001637`]

Out[6]= 2.53116

In[7]:= Sqrt[2.53116 / 10]
        квадратный корень

Out[7]= 0.503106

In[8]:= 0.503106 / 26 * 100

Out[8]= 1.93502

In[9]:= 0.503106 / 2000 * 100

Out[9]= 0.0251553
```

Лістинг 3

Функція InterpolatingPolynomial

Для інтерполяції поліноміальних функцій використовують функцією $InterpolatingPolynomial$, яка має такий формат:

$InterpolatingPolynomial\ [z, x]$,

де z —матриця вихідних даних,

x —аргумент функції z.

Приклад використання функції $InterpolatingPolynomial\ [z, x]$ продемонструємо на прикладі даних, наведених у табл. 3.

Таблиця 3. Табличні дані

X	1	2	3	4	5
Y	1	8	27	64	125

Методика використання функції показана у лістингу 4. Згідно з лістингом функція інтерполяції має вид $y = x^3$. Розв'язок є точним. Для спрощення виразу використано функцію $Simplify$.

```
In[30]:= z = {{1, 1}, {2, 8}, {3, 27}, {4, 64}, {5, 125}};
         InterpolatingPolynomial[z, x]

Out[31]= 1 + (-1 + x) (7 + (-2 + x) (3 + x))

In[32]:= Simplify[%]

Out[32]= x^3
```

Лістинг 4

Функція $InterpolatingPolynomial\,[z, x]$ розв'язує задачі інтерполяції також у випадку не рівновіддалених вузлів. Елементами вектора z може бути набір функцій $f_1(x), f_2(x), \ldots$. Тоді розв'язок видаватиметься у вигляді багаточлена зі зберіганням вихідного виду виразу $f_1(x), f_2(x), \ldots$, лістинг 4.

```
In[33]:= z = {1, Sin[x], 1 / x, Exp[-x]};
         z1 = InterpolatingPolynomial[z, x]
```

$$\text{Out[34]=}\ 1 + (-1 + x)\left(-1 + \mathrm{Sin}[x] + (-2 + x)\left(\frac{1}{2}\left(1 + \frac{1}{x} - 2\,\mathrm{Sin}[x]\right) + \frac{1}{3}(-3 + x)\left(\frac{1}{2}\left(e^{-x} - \frac{2}{x} + \mathrm{Sin}[x]\right) + \frac{1}{2}\left(-1 - \frac{1}{x} + 2\,\mathrm{Sin}[x]\right)\right)\right)\right)$$

```
In[35]:= Table[z1, {x, 1, 4}]
         N[%]
```

$$\text{Out[35]=}\ \left\{1,\ \mathrm{Sin}[2],\ 1 + 2\left(-1 + \frac{1}{2}\left(\frac{4}{3} - 2\,\mathrm{Sin}[3]\right) + \mathrm{Sin}[3]\right),\ 1 + 3\left(-1 + \mathrm{Sin}[4] + 2\left(\frac{1}{2}\left(\frac{5}{4} - 2\,\mathrm{Sin}[4]\right) + \frac{1}{3}\left(\frac{1}{2}\left(-\frac{1}{2} + \frac{1}{e^4} + \mathrm{Sin}[4]\right) + \frac{1}{2}\left(-\frac{5}{4} + 2\,\mathrm{Sin}[4]\right)\right)\right)\right)\right\}$$

```
Out[36]= {1., 0.909297, 0.333333, 0.0183156}
```

Лістинг 5

Значення функції z показані в табл. 4.

Таблиця 4. Значення функції

x	1	2	3	4
y	1	$sinx$	$1/x$	e^{-x}
$y(x)$	1	0,909297	0,333333	0,0183156

Функція Interpolation[data]

Ця функція дозволяє розв'язувати задачу інтерполяції над даними (data) у діапазоні аргументів, які задані цими даними. При цьому функція апроксимації користувачу невідома. Дані наводяться у вигляді матриці функції $y = f(x)$. При введені цієї функції Mathematica видає не функцію інтерполяції, а нову функцію:

$InterpolatingFunction[\{\{x_H, x_K\}\}, <>]$,

де x_H, x_K –діапазон аргументів функції інтерполяції.

Якщо тепер ввести значення аргументу з діапазону $x_H - x_K$, то відповідь буде значенням функції при заданому значенні аргументу.

Розглянемо методику використання функції на прикладі. Таблична функція $y = f(x)$ наведена в табл. 5. Необхідно визначити значення функції при x= 5.8 і x = 18.5 та перевірити достовірність отриманих розв'язків.

Таблиця 5. Функція $y = f(x)$ в табличній формі

X	2	3	8	12	20
Y	1	2,5	4,6	3,2	1,6

Розв'язання задачі наведено у лістингу 5.

```
In[37]:= f = {{2, 1}, {3, 2.5}, {8, 4.6}, {12, 3.2}, {20, 1.6}};
         y = Interpolation[f]

Out[38]= InterpolatingFunction[ Domain: {{2., 20.}}
                                 Output: scalar ]

In[39]:= y[5.8]

Out[39]= 4.56372

In[40]:= y[18.5]

Out[40]= 1.18763

In[41]:= Table[y[{2, 3, 8, 12, 20}]]

Out[41]= {1., 2.5, 4.6, 3.2, 1.6}
```

Лістинг 6

Відповідно до лістингу 6 функція $Interpolation[data]$ інтерполює дані точним методом у вузлах. Результатами табуляції є точні значення функції у вузлах інтерполяції.

Інтерполяція нелінійними функціями

Якщо інтерполяційна функція є нелінійною, то для визначення її коефіцієнтів точним методом у вузлах використовуються два способа:

- створення й розв'язування системи нелінійних рівнянь;

- лінеаризація нелінійної інтерполяційної функції шляхом перетворення координат.

Розглянемо обидва способи.

Спосіб 1. Розв'язування системи нелінійних рівнянь

Методика інтерполяції цим способом полягає в розв'язувані системи нелінійних рівнянь. Продемонструє її на основі даних, наведених у табл. 6.

Таблиця 6. Функція $y = f(x)$ у табличній формі

X	1	2	3	4	5	6	7	8
У	7,6	16	33	71	156	341	750	1650

Інтерполяційна функція має аналітичний вираз $y = ab^x + c$. Необхідно визначити невідомі a, b, c та похибку функції інтерполяції. Оберемо три наступні координати функції: (1,7.6), (4,71), (7, 750) і складемо систему рівнянь:

$$ab^1 + c = 7{,}6$$
$$ab^4 + c = 71$$
$$ab^7 + c = 750$$

Розв'яжемо цю систему нелінійних рівнянь за допомогою функції FindRoot за наступними початковими значеннями невідомих: $a = 2{,}5, b = 3, c = 1{,}5$. Розв'язок наведено у лістингу 7.

```
In[43]:= f = {a b + c == 7.6, a b^4 + c == 71, a b^7 + c == 750}
         FindRoot[f, {a, 2.5}, {b, 3}, {c, 1.5}]

Out[43]= {a b + c == 7.6, a b^4 + c == 71, a b^7 + c == 750}

Out[44]= {a -> 2.96224, b -> 2.20425, c -> 1.0705}

In[45]:= y = 2.96224 * 2.20425^x + 1.0705
         Table[y, {x, 1, 8}]

Out[45]= 1.0705 + 2.96224 × 2.20425^x

Out[46]= {7.60002, 15.4632, 32.7956,
          71.0005, 155.214, 340.841, 750.009, 1651.92}

In[47]:= v1 = {7.5, 16, 33, 71, 156, 341, 750, 1650};
         v2 = {7.6, 15.4632, 32.7956, 71, 155.214, 340.841, 750, 1651.92};
         (v1 - v2) ^2

Out[49]= {0.01, 0.288154, 0.0417794, 0, 0.617796, 0.025281, 0, 3.6864}

In[50]:= Plus[0.009999999999999929`, 0.28815423999999945`,
          0.04177935999999988`, 0, 0.6177960000000021`,
          0.025280999999997396`, 0, 3.6864000000002792`]

Out[50]= 4.66941

In[51]:= Sqrt[Out[50] / 8]

Out[51]= 0.763987

In[52]:= Out[51] / 7.6 100

Out[52]= 10.0525
```

Лістинг 7

З лістингу 7 видно, що математичною моделлю об'єкту є наступна функція інтерполяції:

$y = 2.96 \cdot\ 2.2^{x} + 1.07.$

Абсолютна й відносна середньоквадратична похибка обчислені відповідно до формули (1). Відносна похибка дорівнює 10%.

Спосіб 2. Лінеаризація нелінійної функції

У системі Mathematica апроксимація нелінійними функціями може бути зведена до розв'язування лінійних рівнянь шляхом перетворення координат.

Вирівнювання функцій здійснюється шляхом їх перетворення у лінійну функцію методом заміни змінних. Найбільш просто ці перетворення здійснюються за умови використання степеневих, логарифмічних, дробово-лінійних, показникових функцій. Методика інтерполяції методом лінеаризації нелінійних функцій здійснюється при виконанні наступних дій:

1. Перетворення функції інтерполяції до лінійної форми.
2. Перетворення матриці вихідних даних у матрицю нових змінних.
3. Утворення системи лінійних рівнянь.
4. Розв'язування системи лінійних рівнянь.
5. Утворення функції інтерполяції.
6. Перевірка адекватності отриманої моделі.

Розглянемо методику на прикладі даних, наведених у табл. 7.

Таблиця 7. Функція $y = f(x)$ в табличній формі

X	1	3	5	7	9	11	13	15
Y	2,5	7,8	18,7	28,5	39	50	61,7	73,8

Необхідно знайти математичну модель досліджуваного об'єкту, якщо відомо, що функція інтерполяції є показниковою функцією $y = ax^{b}$. Розв'язок задачі показано у лістингу 8.

```
In[58]:= f1 = {1., 3., 5., 7., 9., 11., 13., 15.};
         f2 = {2.5, 7.8, 18.7, 28.5, 39., 50., 61.7, 73.8};
         Log[f1]
Out[58]= {0., 1.09861, 1.60944, 1.94591, 2.19722, 2.3979, 2.56495, 2.70805}

In[59]:= Log[f2]
Out[59]= {0.916291, 2.05412, 2.92852,
          3.3499, 3.66356, 3.91202, 4.12228, 4.30136}

In[60]:= Solve[{2.054 == A + 1.09861 b, 3.912 == A + 2.398 b}, {A, b}]
Out[60]= {{A → 0.483096, b → 1.4299}}

In[61]:= a = Exp[0.483096]
Out[61]= 1.62109

In[62]:= y = 1.62 x^1.43
         Table[{x, y}, {x, 1, 15, 2}]
Out[62]= 1.62 x^1.43

Out[63]= {{1, 1.62}, {3, 7.79468}, {5, 16.1824}, {7, 26.1821},
          {9, 37.5044}, {11, 49.9697}, {13, 63.4533}, {15, 77.862}}
```

Лістинг 8

У перших двох рядках лістингу 8 знаходяться вектори вихідних даних із ім'ям $f1$ і $f2$. За ними йдуть ті самі вектори в логарифмічному маштабі. Для обчислення коефіцієнтів a і b лінійної функції $lny = lna + blnx$. За вихідні дані взято координати (3,7), (11,50), які в логарифмічному маштабі мають значення: (1.09861, 2.05412), (2.3979, 3.91202). Тоді система рівнянь має вигляд:

$2.05412 = A \cdot 1.098616^b$;

$3.91202 = A \cdot 2.3979^b$,

де $A = lna$.

Система розв'язується за допомогою функції $Solve$. У результаті розв'язування отримані наступні значення коефіцієнтів: $a = 0.483096$, $b = 1.4299$. Тоді $a = e^A = 1.62109$, а функція інтерполяції має вигляд: $y = 1.62x^{1.43}$. Коефіцієнти a і b функції інтерполяції округлені до двох значущих цифр після крапки. Адекватність моделі доведена табуляцією функції інтерполяції за допомогою функції $Table$. Порівнявши результати табуляції з вихідними даними, можна зробити висновки, що задача розв'язана правильно (значення функцій практично збігаються у вузлах інтерполяції $x = 3$ і $x = 11$), а функція інтерполяції є математичною моделлю об'єкту, що вивчається. З метою порівняння результатів двох способів інтерполяції була розв'язана задача інтерполяції нелінійної функції $y = ax^b$. Були обрані ті самі вузли інтерполяції $x = 3$ і $x = 11$ й розв'язана система нелінійних рівнянь за допомогою функції $FindRoot$ за початкових наближеннях $a_0 = 2,\ b_0 = 1$. Розв'язок має вигляд:

```
In[67]:= f = {7.8 == a 3^b, 50 == a 11^b}
         FindRoot[f, {a, 2}, {b, 1}]
```

Out[67]= $\{7.8 == 3^b\,a,\ 50 == 11^b\,a\}$

Out[68]= $\{a \to 1.62121,\ b \to 1.42994\}$

Відповідно до розв’язку, коефіцієнти функції інтерполяції практично збігаються у випадку лінеаризації рівняння.

Інтерполяція, наближена у вузлах

Інтерполяція, наближена у вузлах (апроксимація), здійснюється за критерієм мінімуму середньоквадратичної похибки (метод найменших квадратів). Реалізується системою Mathematica за допомогою функції Fit. Функція Fit має вигляд:

$Fit[\{\{M\}\}, \{X\}, x]$,

де M —матриця вихідних даних,

X — перелік базисних змінних,

x — аргумент функції.

Методика інтерполяції за допомогою функції $Fit[\{\{M\}\}, \{X\}, x]$ реалізується при виконанні наступних дій:

1. Введення матриці вихідних даних із присвоєнням їй унікальної назви, наприклад, M.
2. Введення базисних змінних *X*.
3. Введення функції $Fit[\{\{M\}\}, \{X\}, x]$.

Приклад розв’язування задач інтерполяції покажемо на даних, наведених у табл. 8.

Таблиця 8. Функція $y = f(x)$ в табличній формі

x	1	3	4	7	10
Y	3,5	6,7	4,2	2,8	1,2

Необхідно знайти функцію інтерполяції в базисі $a, x, x^2, \frac{x}{1+x}, e^x$. У даному прикладі функція Fit матиме вигляд:

$$Fit[\{\{1,3.5\},\{3,6.7\},\{4,4.2\},\{7,2.8\},\{10,1.2\}\},\left\{a, x, x^2, \frac{x}{1+x}, Exp[x]\right\}, x]$$

Розв’язок показано у лістингу 9.

```
In[70]:= M = {{1, 3.5}, {3, 6.7}, {4, 4.2}, {7, 2.8}, {10, 1.2}}
         X = {1, x, x^2, x / (1 + x), Exp[x]}
         Fit[M, X, x]
```

Out[70]= $\{\{1, 3.5\}, \{3, 6.7\}, \{4, 4.2\}, \{7, 2.8\}, \{10, 1.2\}\}$

Out[71]= $\left\{1, x, x^2, \frac{x}{1+x}, e^x\right\}$

Out[72]= $-33.1913 - 0.00090638\,e^x - 16.2137\,x + 1.22525\,x^2 + \frac{103.364\,x}{1+x}$

Лістинг 9

Паде апроксимація

Апроксимацію Паде використовують для інтерполяції функції, що задана в аналітичному вигляді дробово-раціональною функцією.

Функція Паде має вигляд:

$PadeApproximant[f(x),\{x,r,\{n_1,n_2\}\}]$;

де $f(x)$ −функція, що задана в аналітичному вигляді,
x −аргумент функції $f(x)$,
r −точка, поблизу якої справедлива функція апроксимації,
n_1 −степінь багаточлена чисельника,
n_2 −степінь багаточлена знаменника.

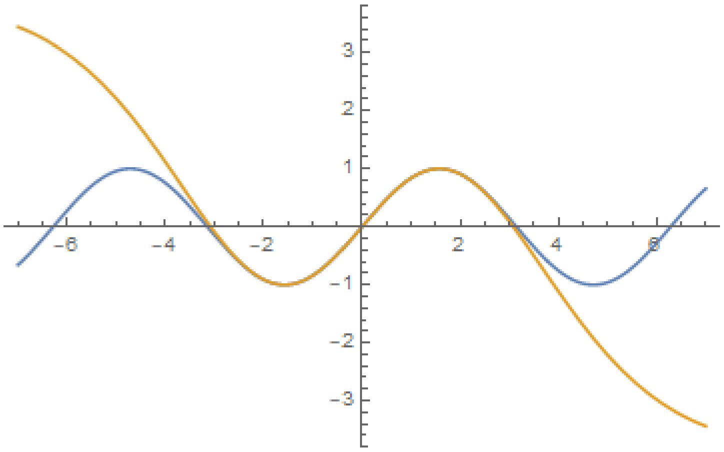
Виконаємо Паде-апроксимацію функції $y = sinx$ поблизу $x = 0$ при $n_1 = 3$ і $n_2 = 4$. Процедура апроксимації в середовищі Mathematica наведена у лістингу 10.

```
In[75]:= f = Sin[x]
Out[75]= Sin[x]
In[77]:= PadeApproximant[f, {x, 0, {3, 4}}]
```

Out[77]= $$\frac{x - \frac{31 x^3}{294}}{1 + \frac{3 x^2}{49} + \frac{11 x^4}{5880}}$$

```
In[78]:= Plot[{f, Out[77]}, {x, -7, 7}]
```

Out[78]=



Лістинг 10

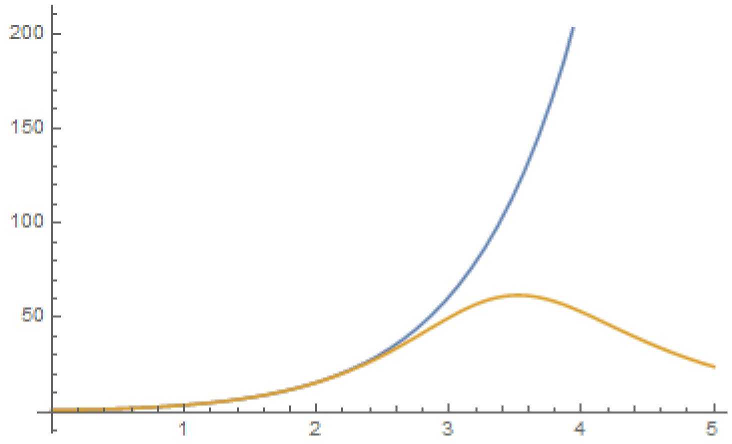
Відповідно до лістингу 10 апроксимація виконана правильно. У діапазоні від $x = -3$ до $x = 3$ функції збігаються. Апроксимація Паде не має обмежень на вигляд вихідної функції, що показано у наступному прикладі. Необхідно виконати Паде-апроксимацію функції $y = xe^x + 1$ поблизу $x = 0$ при $n_1 = 3$ і $n_2 = 4$. Переконатися у певності апроксимації шляхом побудови графіків. Результат апроксимації наведено у лістингу 11.

```
In[79]:= f1 = x Exp[x] + 1
Out[79]= 1 + e^x x
In[80]:= PadeApproximant[f1, {x, 0, {3, 4}}]
```

Out[80]=
$$\frac{1+\frac{x}{4}+\frac{x^2}{2}-\frac{11x^3}{240}}{1-\frac{3x}{4}+\frac{x^2}{4}-\frac{11x^3}{240}+\frac{x^4}{240}}$$

```
In[81]:= Plot[{f1, Out[80]}, {x, 0, 5}]
```

Out[81]=



Лістинг 11

Сплайн-інтерполяція

Функція реалізації сплайн-інтерполяції розміщена у підпакеті SplineFit пакету NumericalMath. Інтерполяція кубічними сплайнами забезпечує високу точність математичної моделі. Перед використання необхідно підключити цей пакет наступним рядком: << Splines` або Needs["Splines`"]. Функція має такий формат:

SplineFit[F, type],

Де F – дані, представлені у вигляді матриці,

type - тип апроксимації, по замовчуванню – апроксимація кубічними сплайнами (Cubic). Також можлива апроксимація сплайнами Bezier та CompositeBezier.

Розглянемо використання функції на прикладі матриці:

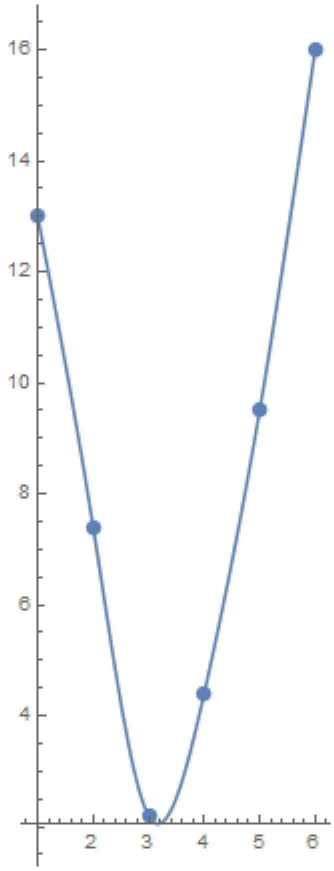
M=((1,13), (2, 7.4), (3, 2.2), (4, 4.4), (5, 9.5), (6, 16)).

Необхідно отримати математичну модель функції $y = f(x)$, використовуючи кубічну сплайн-інтерполяцію. Достовірність рішення перевірити графічно.

Розв'язок приведено у лістингу 12.

```
In[82]:= M = {{1, 13}, {2, 7.4}, {3, 2.2}, {4, 4.4}, {5, 9.5}, {6, 16}}
Out[82]= {{1, 13}, {2, 7.4}, {3, 2.2}, {4, 4.4}, {5, 9.5}, {6, 16}}

In[93]:= << Splines`
         Z1 = SplineFit[M, Cubic]
Out[94]= SplineFunction[Cubic, {0., 5.}, <>]

In[95]:= Z1[0]
Out[95]= {1, 13}

In[96]:= Z1[1]
         Z1[5]
Out[96]= {2., 7.4}

Out[97]= {6., 16.}

In[98]:= Z1[3.53]
Out[98]= {4.53, 6.92269}

In[102]:= r1 = ParametricPlot[Z1[x], {x, 0, 5}];
          r2 = ListPlot[M, PlotStyle → {PointSize[0.05]}];
          Show[r1, r2]
```

Out[104]=



Лістинг 12

З лістингу 11 видно, що розв’язок отриманий не в явному вигляді, тобто аналітичний вираз функції інтерполяції відсутній. Це є суттєвим недоліком функції SplineFit[F, type]. З графіку можна зробити висновок, що інтерполяція виконана достатньо точно.

Хід роботи:

1. Протабулювати функції $y_1(x)$ з кроком Δx_1, $y_2(x)$ з кроком Δx_2 у заданому діапазоні згідно з варіантом, наведеним у табл. 1.

Таблиця. 1. Варіанти функцій

Вар.	Функція	Δx	Діап.	Вар.	Функція	Δx	Діап.
1,11	$y_1 = 0,1x^5 - x^4 - 2x^2 + 3x + 5$	2	-5:11	6,16	$y_1 = 0,01x^7 - x^5 + 3x^3 + 4x^2 + 2$	1	0:10
	$y_2 = \ln(5x)$	1	1:10		$y_2 = -1/3x^2$	1	1:10
2,12	$y_1 = 0,5x^4 - 2x^3 + 6x - 3$	1	-3:5	7,17	$y_1 = -0,06x^6 + x^5 - 4x^3 + x^2 + 1$	2	0:20
	$y_2 = e^{-5x}$	5	0:30		$y_2 = \cosh(3x)$	1	-3:3
3, 13	$y_1 = 0,1x^6 - 3x^5 + 60x - 6$	1	-3:4	8,18	$y_1 = -0,06x^6 + x^5 - 4x^3 + x^2 + 1$	0,5	-3:3
	$y_2 = 10e^{-x^2}$	1	-5:5		$y_2 = \sinh(3x)$	1	-3:3
4,14	$y_1 = -0,1x^5 - 3x^4 + 20x^2 - 6$	1	-3:3	9,19	$y_1 = x^8 - x^3 - 1000x^2 + x + 5$	0,5	-3:3
	$y_2 = sin(3x)$	1	-5:5		$y_2 = \ln(5x) + x$	1	1:10
5,15	$y_1 = x^4 - 15x^3 + 10x^2 + 5x + 12$	2	-5:15	10,20	$y_1 = -x^4 + 3x^3 + 4x^2 + 5x + 5$	0,5	-2:4
	$y_2 = cos(3x)$	1	-5:5		$y_2 = 10e^{-x^2} + x^2$	0,5	0:4

2. Використовуючи масиви табульованих значень провести інтерполяцію функціями:
   - InterpolatingPolynomial;
   - Fit[{{M}}, {X}, x];
   - Pade[f(x), {x, r, n_1, n_2}];
   - SplineFit[F, type].
3. Розрахувати відносну похибку інтерполяції за умови використання кожної з функцій і занести дані у таблицю.
4. Зробити висновки стосовно особливостей використання функцій інтерполяції.

Контрольні запитання

1. Опишіть особливості використання методів інтерполяції, точної у вузлах, і інтерполяції, наближеною у вузлах.
2. Опишіть універсальний метод інтерполяції функцій, точної у вузлах.
3. Вкажіть яким чином оцінюють точність методів інтерполяції.
4. Зазначте особливості використання функції InterpolatingPolynomial[].
5. Зазначте особливості використання функції *Функція* Interpolation[].

6. Опишіть методику інтерполяції методом лінеаризації нелінійних функцій.
7. Вкажіть функції, які використовують для інтерполяції, наближеної у вузлах.
8. Назвіть методи інтерполяції нелінійними функціями.
9. Перелічіть параметри функції паде апроксимації.
10. Вкажіть, яким на практиці є співвідношення між степенем інтерполяційного поліному n і кількістю точок даних функції N.

Лабораторна № 7
Основи програмування в середовищі Mathematica

Мета: отримання навичок програмування в середовищі Mathematica.

Короткі теоретичні відомості
Mathematica як система програмування
Поняття про вхідну мову системи і мов уреалізації

В системі Mathematica можливо вирішити багато задач без використання програмування. Однак всі засоби системи (алфавіт, літери, цифри, оператори та спеціальні знаки) є частиною об’єктно-орієнтованої мови програмування надвисокого рівня. За своїми можливостями у виконанні математичних та науково-технічних обчислень ця мова має значні переваги у порівнянні зі звичайними мовами програмування – Паскаль, C, C++.

Можливості мови програмування системи Mathematica

Система Mathematica містить велику кількість функцій , серед яких чимало таких , які реалізують математичні перетворення і сучасні обчислювальні методи, як чисельні , так і аналітичні . Мова програмування системи Mathematica є типовим інтерпретатором і не призначений для створення виконуваних файлів. Проте, окремі вирази можна компілювати за допомогою функції Compile, що корисно при необхідності збільшення швидкості рахунків. Мова системи Mathematica дозволяє реалізувати більшість типів програмування: функціональне, структурне, об'єктно-орієнтоване, математичне, логічне, рекурсивне і т.д.

Ядро системи Mathematica має функціональну структуру. Мова системи дозволяє розбивати програми на окремі модулі (блоки), процедури і функції з локальними змінними. Об'єктно-орієнтоване програмування базується на узагальненому понятті об'єкта. В системі Mathematica об'єктами можуть бути математичні вирази, вхідні і вихідні дані, графіки та малюнки, звуки і т.д. З поняттям об'єкта тісно пов'язані три основні властивості: інкапсуляція, спадкування і поліформізм. Всі вони притаманні об'єктам системи Mathematica і не вимагають для своєї реалізації спеціальних засобів. Інкапсуляція означає об'єднання в одному об'єкті як даних, так і методів їх обробки. Спадкування означає, що кожен об'єкт, похідний від інших об'єктів, успадковує їх властивості. Поліформізм - властивість, що дозволяє передати ряду об'єктів повідомлення, яке буде оброблятися кожним об'єктом відповідно до його індивідуальних особливостей.

Мова програмування системи Mathematica спеціально створена для реалізації будь-якого з перерахованих підходів до програмування, а також ряду інших, наприклад рекурентного програмування, за умови використання якого черговий крок обчислень базується на даних, отриманих на попередніх кроках. Можливо і рекурсивне програмування - це коли функція в загальному випадку неодноразово звертається до себе самої. Засоби мови Mathematica дозволяють здійснити і елементи візуально-орієнтованого програмування. Mathematica дозволяє створювати палітри і панелі з різними кнопками, що дозволяють керувати програмою або вводити нові програмні об'єкти.

Структура програмного середовища системи Mathematica

Срткутуру програмного середовища системи Mathematica можна представитив наступному вигляді, зображеному на рис. 1.

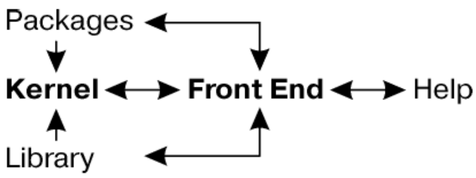


Рис. 1. Структура програмного середовища системи Mathematica

Для орієнтації системи на конкретну машинну платформу служить інтерфейсний процесор **FrontEnd** . Саме він визначає , який вигляд має інтерфейс користувача системи . Центральне місце в системах класу Mathematica займає машинно незалежне ядро математичних операцій - **Kernel** . Воно містить набір операторів і функцій, правил обчислень і перетворень математичних виразів. Ядро зроблено досить компактним для того, щоб будь-яка функція з нього викликалася досить швидко. Для розширення набору функцій служить бібліотека **Library** і набір пакетів розширення **Packages**. Пакети розширень створюють на власній мові програмування систем Mathematica і є головним засобом розширення можливостей системи та їх адаптації до вирішення конкретних класів задач користувача. Крім того, системи мають вбудовану довідкову систему **- Help**.

Основи функціонального програмування

Принцип функціонального програмування передбачає використання під час написання лише функцій. При цьому можливе неодноразове вкладення функцій одна в одну. У ряді випадків, особливо під час символьних перетворень, відбувається взаємна рекурсія функцій, супроводжувана майже необмеженим поглибленням рекурсії і наростанням складності оброблюваних системою виразів. Поняття функції асоціюють з обов'язковим поверненням деякого значення у відповідь на звернення до функції. Повернення функціями деяких значень дозволяє застосовувати їх поряд з операторами для складання математичних виразів. Функції поділяються на внутрішні і функції, задані користувачем.

Функції користувача

Процес створення функції у системі Mathematica подібний до інших мов програмування. Наприклад, функцію для зведення x в ступінь n можна було б визначити так :

powerxn [x , n]: = x ^ n

Однак така функція є непрацездатною. Причиною цього є те, що в системі Mathematica символи x і n є звичайними символами і не можуть сприймати формальні параметри. Для реалізації необхідно використовувати змінні-зразки, що мають після своїх імен символи підкреслення. Зразки можуть бути формальними параметрами функцій і сприймати значення фактичних

параметрів. Таким чином, правильним буде запис функції користувача у вигляді:
powerxn [x_ , n_]: = x ^ n

Розглянемо ще один простий приклад , в якому задана функція scn [x , n], яка обчислює суму синуса в ступені n і косинуса в ступені n, приклади використання якої наведені у лістингу 1:

```
In[443]:= scn[x_, n_] := Sin[x]^n + Cos[x]^n
          scn[1, 2]
Out[444]= Cos[1]^2 + Sin[1]^2

In[445]:= scn[x, 2]
Out[445]= Cos[x]^2 + Sin[x]^2

In[446]:= scn[x, n]
Out[446]= Cos[x]^n + Sin[x]^n
```

Лістинг 1

Функція може складатись з декількох виразів, які об’єднано круглими дужками:
f [x_]: = (t = (1 + x) ^ 2 ; t = Expand [t])
Змінні списку параметрів, після імені яких стоїть знак « _ » , є локальними в тілі функції або процедури з параметрами. На їх місце підставляється фактичне значення відповідного параметра. Приклад, який ілюструє використання локальних змінних показано у лістингу 2:

```
In[462]:=
          f[x_] := (t = (1 + x)^2; t = Expand[t])
          f[a + b]
Out[463]= 1 + 2 a + a^2 + 2 b + 2 a b + b^2

In[464]:= t
Out[464]= 1 + 2 a + a^2 + 2 b + 2 a b + b^2
```

Лістинг 2

Зверніть увагу на те , що змінна t у функції f є глобальною, що пояснює результат останньої операції. Застосування глобальних змінних в тілі функції цілком можливо, але створює так званий побічний ефект, в даному випадку змінює значення глобальної змінної t. Для усунення побічних ефектів необхідно використовувати зразки та інші спеціальні способи створення функцій.

Параметрами функцій можуть бути списки, за умови допустимості їх комбінації. Наприклад у функції powerxn, лістинг 1, припустимо як параметр x використовувати список, а як n – змінну або число, що показано у лістингу 3:

```
In[471]:= powerxn[x_, n_] := x^n
          powerxn[{1, 2, 3, 4, 5}, z]
Out[472]= {1, 2^z, 3^z, 4^z, 5^z}

In[473]:= powerxn[{1, 2, 3, 4, 5}, 2]
Out[473]= {1, 4, 9, 16, 25}
```

Лістинг 3

Після свого створення функції користувача можуть використовуватися за тими ж правилам , що і вбудовані функції.

Чисті функції

Іноді може знадобитися використання деякої функції тільки в момент її створення. Ця функція представлена тільки виразом без імені, що обумовило її назву. Для створення такого об'єкта служить вбудована функція Function, яку використовують у вигляді :

- Function [body] – створює чисту функцію з тілом body;
- Function [{ x}, body] – створює чисту функцію параметра x з тілом body;
- Function [{ x1, x2, ...}, body] – створює чисту функцію ряду параметрів x1, x2 , ... з тілом body.

Для обчислення створеної таким чином функції після неї записують список параметрів у квадратних дужках. Як приклад у лістингу 4 записано код чистої функції підесення до степеня:

```
In[474]:= Function[{x, n}, x^n]
Out[474]= Function[{x, n}, x^n]

In[475]:= %[2, 3]
Out[475]= 8
```

Лістинг 4

Чисту функцію можна легко перетворити на звичайну функцію користувача, що показує приклад лістинга 5:

```
In[476]:= fun = Function[{x, n}, x^n]
Out[476]= Function[{x, n}, x^n]

In[477]:= fun[2, 3]
Out[477]= 8
```

Лістинг 5

Анонімні функції

Гранично компактну форму задання функцій мають так звані анонімні функції. Вони не мають ні назви, ні звичайного визначення, їх записують виразами спеціального виду. У цьому виразі замість змінних використовують позначення # (для однієї змінної) , # 1 , # 2 , ... для ряду змінних. Завершують тіло функції символом & . Якщо необхідно розрахувати значення функції, то після її запису в квадратних дужках вказують список фактичних параметрів. Приклад застосування анонімної функції показано у лістингу 6:

```
In[478]:= #1^#2 &[2, 3]
Out[478]= 8

In[481]:= #1^#2 &[y, z]
Out[481]= y^z
```

Лістинг 6

За допомогою анонімних функцій неважко створити звичайні функції користувача – лістинг 7:

```
In[482]:= f[x_, y_] = #1^#2 &[x, y]
Out[482]= x^y

In[483]:= f[2, 3]
Out[483]= 8
```

Лістинг 7

Суперпозиція функцій

Функціональне програмування передбачає використання суперпозиції функцій. Для її реалізації використовують функції:

- Nest[expr, x, n] – застосовує вираз (функцію) до заданого аргументу x n разів.
- NestList[f , x, n] – повертає список використань функції f до заданого аргументу x n +1 раз .
- Fold [f, x, list] – повертає наступний елемент у FoldList[f , x, list].
- FoldList [f, x, {a, b, ... }] – повертає {x, f [x, a], f [f [x, a], b], j}.
- ComposeList[{f1, f2, ...}, x] – генерує список у формі {x , a[x], a[a[x]], ...}.

Функції FixedPoint і Cath

У функціональному програмуванні замість циклів , описаних далі , можна використовувати наступну функція:

- FixedPoint [f, expr] – обчислює expr і застосовує до нього вираз f , поки результат не повториться.
- FixedPoint [f , expr , SameTest_ > comp] - обчислює expr і застосовує до нього вираз f, поки два наступних результати не будуть істинними.

Приклад застосування функції FixedPoint наведено у лістингу 8.

```
In[484]:= FixedPoint[Function[t, Print[t]; Floor[t / 2]], 27]
27
13
6
3
1
0
Out[484]= 0
```

Лістинг 8

Останній результат 0 виводиться в окремій (нумерованій) комірці виведення і означає завершення процесу ітерацій ділення t на 2. Ланцюговий дріб за допомогою функції Nest можна створити за допомогою таких аргументів:

```
In[485]:= Nest[Function[t, 1/(1+t)], y, 3]
```

$$\mathrm{Out[485]=}\ \frac{1}{1+\frac{1}{1+\frac{1}{1+y}}}$$

Лістинг 9

Ще одна функція такого роду - Catch []:

- Catch [expr] – обчислює expr, до першого виконання функції Throw[value], після чого повертає value .

- Catch [expr, form] – обчислює expr до першого виконання функції Throw[value, tag], після чого повертає value .

- Catch[expr, form, f] – повертає f[value, tag] замість value .

Реалізація рекурсивних та рекурентних алгоритмів

Важливе місце у вирішенні багатьох математичних задач займає реалізація рекурсивних та рекурентних алгоритмів. Розглянемо типовий приклад реалізації ітераційного рекурентного алгоритму обчислення квадратного кореня з виразу f(x) при початковому значенні x0 = a , за такими формулами метода Ньютона:

$x_0 = a$ і $x_n = x_{n-1} - f(x_{n-1})/f'(x_{n-1})$.

Цю функцію можна записати таким чином:

newtoniter[f_, x0_, n_]: = Nest[(# - f [#] / f '[#]) &, N[x0], n]

Тоді обчислення кореня з виразу e^x - 2 з початковим наближенням x_0 = 0.5 і кількістю ітерацій n можна організувати за допомогою функцій Nest[] і NestList[], що показано у лістингу 10:

```
In[493]:= newtoniter[f_, x0_, n_] := Nest[(# - f[#] / f'[#]) &, N[x0], n]
          newtoniter[Function[{x}, Exp[x] - 2.0], 0.5, 5]

Out[494]= 0.693147

In[492]:= newtoniter[Function[{x}, Exp[x] - 2.0], 0.5, #] & /@ Range[5]

Out[492]= {0.713061, 0.693344, 0.693147, 0.693147, 0.693147}

In[496]:= newtoniter1[f_, x0_, n_] :=
           NestList[(# - f[#] / f'[#]) &, N[x0], n]
          newtoniter1[Function[{x}, Exp[x] - 2.0], 0.5, 5]

Out[497]= {0.5, 0.713061, 0.693344, 0.693147, 0.693147, 0.693147}
```

Лістинг 10

У першому випадку повернено останній результат , а в інших – всі проміжні. Функція FixedPoint дозволяє здійснювати ітерації до тих пір, поки результат не перестане змінюватися (з машинною точністю). Це ілюструє приклад, наведений у лістингу 11.

```
In[509]:= newtonfp[f_, init_, options_] :=
            FixedPoint[#1 - f[#1] / f'[#1] &, init, options];
          newtonfp[Function[{x}, Exp[x] - 2.0], 0.5, 5]

Out[510]= 0.693147
```

Лістинг 11

Рекурсивні алгоритми використовують обчислення, у яких у тілі функції відбувається звернення до неї ж самої. Mathematica допускає таку можливість. Типовий приклад цього - обчислення факторіала по формулі N! = N * (N-1)! .

Основи процедурного програмування

В основі процедурного програмування лежить поняття процедури як закінченого програмного модуля і типових засобів керування: циклів, умовних та безумовних переходів і т.д. Хоча, програмування систем Mathematica і в цьому випадку залишається функціональним, оскільки елементи процедурного програмування задані в вигляді функцій. Однак з'являється можливість застосування традиційних засобів програмування – умовних виразів, циклів, процедур і т.д.

Процедури є повністю самостійними програмними модулями , мають свій ідентифікатор і виконують деяку послідовність операцій. Вони можуть бути описані в одному рядку з використанням в якості роздільника символа «;» (крапка з комою), наприклад:

r=(1+x)^2;r=Expand[r];r-1

Ця процедура повертає такий символьний вираз:

Expand[(1+x)^2] - 1

Для створення повноцінних процедур і функцій використовують структуру Block у двох модифікаціях:

Block[{x, y, ...}, procedure] – процедура з декларацією списку локальних змінних x,y,…

Block[{x = x0, y=y0, ...}, procedure] – процедура з декларацією списку локальних змінних x,y,… з початковими значеннями.

Приклад використання структури Block наведено у лістингу 12:

```
In[511]:= g[x_] := Block[{u}, u = (1 + x)^2; u = Expand[u]]
          g[a + b]

Out[512]= 1 + 2 a + a^2 + 2 b + 2 a b + b^2

In[513]:= u

Out[513]= u

In[514]:= u = 123456; g[2]

Out[514]= 9

In[515]:= u

Out[515]= 123456
```

Лістинг 12

Зверніть увагу, що змінна u, яку використовують у тілі базової структури, є локальною, і присвоєння їй символьного виразу (1+ x) ^ 2 в тілі блоку ігнорується поза блоком. Якщо змінна u до застосування у функції була не визначена,то вона так і залишається невизначеною. А якщо вона мала до цього деяке значення (наприклад, 123456 у нашому випадку), то і після виходу з процедури вона матиме це значення.

Організація циклів

Цикли типу Do

Цикли цього типу мають декілька модифікацій:

- Do[expr, {imax}] – розраховує вираз expr imax разів.
- Do[expr, {i, imax}] – розраховує вираз expr зі змінною i, яка почергово приймає значення від 1 до imax з кроком 1.
- Do[expr, {i, imin, imax}] – розраховує вираз expr зі змінною i, яка почергово приймає значення від imin до imax з кроком 1.
- Do[expr, {i, imin, imax, di}] – розраховує вираз expr зі змінною i, яка почергово приймає значення від imin до imax з кроком di.
- Do[expr, {i, imin, imax}, {j, jmin, jmax}, ...] – розраховує вираз expr з використаннями ряду вкладених циклів зі змінними j, i і т.д.

Приклади організації циклу Do і його виконання наведені у лістингу 13:

```
In[516]:= Do[Print["Hello"], {5}]
          Hello
          Hello
          Hello
          Hello
          Hello

In[517]:= Do[Print[i], {i, 3}]
          1
          2
          3

In[518]:= Do[Print[i], {i, 5, 8}]
          5
          6
          7
          8

In[519]:= Do[Print[i], {i, 0, 1, 0.25}]
          0.
          0.25
          0.5
          0.75
          1.
```

Лістинг 13

Змінна i в тілі циклу – ітератор – є локальною. Вся програма з циклом зберігається в одній комірці. В лістингу 14 показано процедуру з циклом Do, що обчислює n-не число Фібоначчі:

```
In[520]:= fibonacci[(n_Integer)?Positive] :=
           Module[{fn1 = 1, fn2 = 0}, Do[{fn1, fn2} = {fn1 + fn2, fn1}, {n - 1}];
            fn1]
```

Лістинг 14

Зверніть увагу на застосування в цьому прикладі функції Module. Вона створює програмний модуль з локальними змінними (у нашому випадку fn1 і fn2), в якому організовано рекуррентное обчислення чисел Фібоначчі. У лістингу 15 показано застосування циклу Do для створення ланцюгового дробу:

```
In[522]:= x = y;
          Do[x = 1 / (1 + k x), {k, 2, 8, 2}]; x
```

Out[523]= $$\frac{1}{1+\frac{8}{1+\frac{6}{1+\frac{4}{1+2\,y}}}}$$

Лістинг 15

Цикли типу For

Інший вид циклу For реалізується функцією:

For [start, test, incr, body]

У ній змінній циклу спочатку присвоюється значення start, потім циклічно змінюється від цього значення до значення body з кроком incr; і так до тих пір, поки умова test буде істинною. Коли умова стане хибною, цикл закінчується. У прикладі лістингу 16 показано просту програму з циклом For і результат її виконання:

```
In[526]:= Print["i x"]
          For[x = 0;
           i = 0, i < 4, i++[x += 5 * i, Print[i, " ", x]]]
          i x
          1 5
          2 15
          3 30
          4 50
In[528]:= Return[x]
Out[528]= Return[50]
```

Лістинг 16

Програма, наведена вище, дозволяє спостерігати за зміною значень змінної циклу *i* і змінної x, що одержує за кожен цикл прирощення рівне 5*i. У кінці документа показаний приклад на використання функції повернення значень Return [x]. У циклі For використовують глобальні змінні, тому необхідний контроль за їх використанням.

Цикли типу While

Форма запису циклу:

While [test, expr] .

У цьому типі циклу розраховується значення expr до тих пір, поки умова test істинна. Нижче розглянуто приклад з організації та використання циклу While.

```
In[535]:= i = 1; x = 1;
          Print["i x"]
          While[i < 5, i += 1;
           x += 2 * i;
           Print[i, " ", N[x]]]

          i x
          2 5.
          3 11.
          4 19.
          5 29.
```

Лістинг 17

Апарат локальних змінних в цьому типі циклів не використовується.

Директиви переривання і продовження циклів

У зазначених типах циклів і в інших структурах можна використовувати такі директиви-функції:

- Abort [ ] – припиняє обчислення з повідомленням $ Aborted.
- Break [ ] – виконує вихід з тіла циклу або рівня вкладеності програми, що містить даний оператор (цикли типу Do , For і While або тіло оператора - перемикача Swith) . Оператор повертає Null - значення.
- Continue [ ] – задає перехід на наступний крок поточного циклу Do , For абоWhile .
- Interrupt [ ] – перериває обчислення з можливістю їх поновлення за допомогою діалогового вікна.
- Return [ ] – перериває виконання з поверненням значення Null .
- Return [expr] – перериває виконання з поверненням значення виразу expr.

Умовні вирази і безумовні переходи

Функція If

Як і у більшості мов програмування, умовні вирази задаються за допомогою оператора або функції If. Система Mathematica має такі модифікації функцію If:

- If [condition, t, f] – повертає результат t, якщо обчислення умови condition істинне, та f, якщо результат хибний.
- If [condition, t, f, u] - повертає результат u, якщо результат обчислення умови condition не буде ні істинним, ні хибним.

У лістингу 18 показаний опис процедури з циклом Do, вихід з якої організовано за допомогою функції If і директиви переривання Aborted []:

```
In[542]:= x := 1; Print["i x"];
          Do[{If[i == 5, Abort[], None], i += 1;
            x += 2 * i;
            Print[i, " ", N[x]]}, {i, 1, 100}]
          i x
          2 5.
          3 11.
          4 19.
          5 29.
Out[543]= $Aborted

In[545]:= Return[x]
Out[545]= Return[29]
```

Лістинг 18

Аналогічний приклад з використанням функції Break у функції If показано у лістингу 19.

```
In[546]:= x := 1;
          Print["i x"];
          Do[{If[i == 5, Break[], None], i += 1;
            x += 2 * i;
            Print[i, " ", N[x]]}, {i, 1, 100}]
          i x
          2 5.
          3 11.
          4 19.
          5 29.
In[547]:= Return[x]
Out[547]= Return[29]
```

Лістинг 19

У даному випадку ніяких спеціальних повідомлень про вихід з циклу не дається.

Функції - перемикачі

Для організації декількох розгалужень в системі Mathematica використовують оператори – перемикачі Which і Switch :

- Which [$test_1$, $value_1$, $test_2$, $value_2$, ...] - обчислює в порядку проходження кожну умову $test_i$ і повертає значення $value_i$ першої умови $test_i$, яка була істинна.

- Switch [expr , $form_1$, $value_1$, $form_2$, $value_2$, ...] - обчислює значення умови expr , потім порівнює його послідовно з кожним виразом $form_i$ і повертає значення $value_i$, першого збігу виразу $form_i$ з умовою expr.

Приклади використання функції Which наведено у лістингу 20.

```
In[548]:= Which[1 == 2, 1, 2 == 2, 2, 3 == 3, 3]
Out[548]= 2

In[549]:= Which[1 == 2, x, 2 == 2, y, 3 == 3, z]
Out[549]= y
```

Лістинг 20

Приклади, що демонструють використання функції Switch наведено у лістингу 21.

```
In[550]:= Switch[2, 1, a, 2, b, 3, c]
Out[550]= b

In[551]:= Switch[3, 1, a, 2, b, 3, c]
Out[551]= c

In[552]:= Switch[8, 1, a, 2, b, 3, c]
Out[552]= Switch[8, 1, a, 2, b, 3, c]
```

Лістинг 21

Зверніть увагу на останній приклад: за умови неправильного запису першого параметра, запис функції повторюється.

Безумовні переходи

Оператор безумовного переходу Goto [tag] створює перехід до місця програми, позначене міткою Label [tag]. Можливі також форми Goto [expr] і Label [expr], де expr – певний вираз. Використання оператора Goto показано у лістингу 20.

```
In[553]:= (q = 2; Label[start]; Print[q]; q += 2;
           If[q < 7, Goto[start]])
          2
          4
          6
```

Лістинг 22

Тут за допомогою оператора Goto [start] організовано цикл з переходом на мітку Label [start], який виконується , поки значення q менше 7. При цьому q змінюється від початкового значення 2 з кроком 2. Цікавою особливістю мови програмування Mathematica є можливість створення переходів за значенням обчислюваного виразу. Наприклад, Goto [2 + 3] дає перехід до мітки Label [5] або навіть Label [1 + 4], що видно з наступного прикладу:

```
In[636]:= (q = 2; Label[1 + 4]; Print[q]; q += 2;
           If[q < 7, Goto[2 + 3]])
          2
          4
          6
```

Лістинг 23

Хід роботи

1. Написати програму для вирішення задачі згідно з варіантом, наведеним у табл.1.

Вар.	Завдання
1	Для заданого числа k сформувати список з елементів i^k, де i – ціле число в діапазоні 1-10. Побудувати графік за значеннями списку. Розрахувати середнє значення і середньо-квадратичне відхилення всіх елементів масиву.
2	Вводяться два числа i та j. З двох чисел обирають найбільше і використовують його для створення списку. Нехай більшим число є число j. Тоді перший елемент списку рівний числу 1-i/j. Змінній j присвоюють нове значення j=j(1-i/j). Знаходять найбільше число з двох i та j. Нехай більшим число є число j. Тоді другий елемент списку рівний числу 1-i/j і т.д. Написати програму яка формує 10 елементів списку.
3	Розрахувати інтеграл заданої функції в діапазоні x = 0-10, з кроком Δx = 0.1 методом трапецій.
4	Побудувати трикутник Паскаля.
5	Дано цілочисельну матрицю розміру M x N. Знайти кількість її рядків і стовпців, всі елементи яких відмінні один від одного.
6	Дано матрицю розміру M x N. Вивести номер її рядка, який містить максимальну кількість однакових елементів.
7	Дано матрицю розміром M x N. Знайти елемент, який є максимальним у своєму рядку і мінімальним у своєму стовпчику. Якщо такий елемент відсутній вивести 0.
8	Дано набір из N дійсних чисел. Перевірити чи утворюють вони зростаючу послідовність. Якщо утворюють вивести 1, інакше – 0.
9	Написати програму, яка знаходить розклад натурального числа на прості множники. Результатом виконання програми є кількість простих множників N і їх значення у порядку зменшення значення.
10	Написати програму, яка здійснює циклічний зсув елементів списку на k позицій. Якщо k – додатне зсув здійснюють праворуч, інакше – ліворуч.
11	Написати програму, яка змінює порядок розташування елементів списку на зворотній.
12	Написати програму, яка сумує елементи матриці з рядка n_1 до рядка n_2, починаючи зі стовпчика k_1, закінчуючи стовпчиком k_2.
13	Написати програму, яка видаляє з матриці рядок і стовпчик, який містить елемет зі значення k.
14	Знайти позиції елементів масиву, різниця значень яких мінімальна.
15	Дано цілочисельний масив. Знайти найбільшу кількість його однакових елементів.

Контрольні запитання

1. Назвіть можливості мови програмування системи Mathematica.
2. Опишіть структуру програмного середовища системи Mathematica.
3. Зазначте особливості створення функцій користувача.
4. Зазначте особливості функціонального стилю програмування.
5. Зазначте особливості процедурного стилю програмування.
6. Перелічіть функції для створення циклів.
7. Назвіть функції для створення умовних виразів і безумовних переходів.
8. Вкажіть призначення функцій FixedPoint[] і Cath[].
9. Вкажіть функції організації циклів, які мають локальні змінні-ітератори, а які глобальні.
10. Дайте визначення терміну «чиста функція».

Лабораторна робота № 8
Графічні об'єкти введення-виведення

Мета: отримання навичок зі створення графічних об'єктів введення-виведення середовища Mathematica.

Короткі теоретичні відомості

Введення-виведення в системі Mathematica організовано за допомогою інтерфейсного процесору FrontEnd. Додатково у системі реалізовано ряд додактових функцій введення-виведення:

- Input[] – зупиняє роботу системи і повертає значення виразу яке буде введено в діалоговому вікні (використовують для організації діалогового введення);

- Input[«коментар»] – працює аналогічно попередній функції, додатково виводить у діалогове вікно «коментар»;

- InputString[ ] – зчитує данні і зберігає їх у формі символьного рядка;

- InputString[«коментар»] – працює аналогічно попередній функції, додатково виводить у діалогове вікно «коментар»

- StylePrint[expr] – створює в поточному документі нову комірку зі стилем за замовчуванням і заносить в нього вираз expr;

- StylePrint[expr, «style»] – створює у поточному документі нову комірку зі стилем style і заносить в неї вираз expr;

- Print[expr] – виводить на екран дисплея значення виразу expr, разом з функцією Input[] може використовуватись для организації діалогу;

- Print[«prompt», expr] – виводить на екран дисплея текстовий коментар у лапках, а згодом – значення виразу expr.

Цих функцій достатньо для організації найпростішого діалогу з программою. У лістнингу 1 показано найпростіший приклад організації діалогу. У данному випадку обчислюється довжина кола за значенням радіуса R.

```
In[7]:= R = Input["Input radius R"]

Out[7]= 3

In[10]:= L := 2*Pi*R
         Print["Circuit length=", L]

         Circuit length=6 π
```

Лістинг 1

Після виконання функції Input[] з'являється діалогове вікно у центрі екрану. У вікні виводиться запит, який вказаний у лапках як параметр функції Input[]. Після введення потрібного значення (у загальному прикладі це радіус кола) функція Input[] повертає введене значення і воно присвоюється змінній R.

Після цього функція Print[] виводить на екран обчислене значення довжини кола з коротким коментарем.

Формат виведення даних

В програмі Mathematica реалізовано ряд функцій для налаштування формату представлення даних. Найчастіше використовують такі функції:

- AccountingForm[expr] – виодить всі числа, що містяться в виразі expr, у бухгалтерскій формі представлення;
- CForm[expr] – виводить визраз expr у форматі мови C;
- EngineeringForm[expr] – виодводить дійсні числа виразу expr в інженерній формі;
- FortranForm[expr] – виводить вираз expr в формі, принятої для мови Фортран;
- FullForm[expr] – виводить повну форму виразу expr без використання спеціального синтаксису;
- InputForm[expr] – виводить вираз expr у вхідній формі;
- NumberForm[expr, n] – виводить вираз expr у формі дійсного числа з точністю до n цифр;
- OutputForm[expr] – виводить враз expr в стандартній вихідній формі системи Mathematica;
- ScientificForm[expr] – виводить вираз expr у науковому форматі;
- TeXForm[expr] – виводить вираз expr у формі мови TeX, яка орієнтована на верстання текстів з математичними формулами;
- TextForm[expr] – виводить вираз expr в звичайному текстовому форматі;
- TreeForm[expr] – виводить вираз expr з демонстрацією різних рівнів виразу.

У лістингу 2 наведено приклади використання різних форм виведення.

```
In[18]:= AccountingForm[30 * 10^15]
Out[18]//AccountingForm=
         30000000000000000

In[19]:= BaseForm[55434, 16]
Out[19]//BaseForm=
         d88a_16

In[20]:= CForm[x^2 + 3 * x + x]
Out[20]//CForm=
         4*x + Power(x,2)

In[21]:= ColumnForm[{a, b, c}]
Out[21]= a
         b
         c

In[22]:= EngineeringForm[N[12 * 10^29]]
Out[22]//EngineeringForm=
         1.2×10^30

In[23]:= Format[Exp[x^2] / a]
```

$$\text{Out[23]=}\ \frac{e^{x^2}}{a}$$

```
In[24]:= FortranForm[Exp[x]^2 / a]
Out[24]//FortranForm=
         E**(2*x)/a

In[25]:= HoldForm[Exp[x]^2 / a]
```

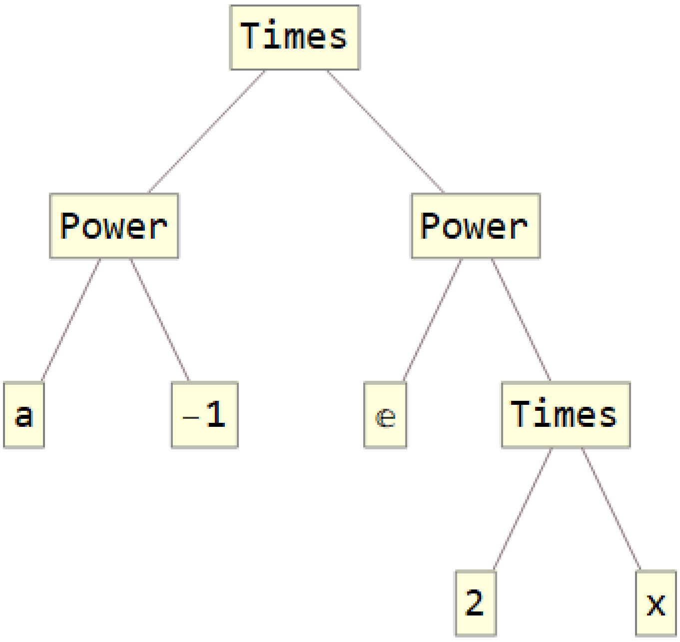
$$\text{Out[25]=}\ \frac{\text{Exp}[x]^2}{a}$$

```
In[26]:= NumberForm[N[Exp[2]], 15]
Out[26]//NumberForm=
         7.38905609893065

In[27]:= OutputForm[Exp[x]^2 / a]
Out[27]//OutputForm=
          2 x
         E
         ----
          a

In[28]:= TeXForm[Exp[x]^2 / a]
Out[28]//TeXForm=
         \frac{e^{2 x}}{a}

In[29]:= ScientificForm[12 * 10^5]
Out[29]//ScientificForm=
         1200000
```

Лістинг 2

У лістингу 3 наведено ще кілька прикладів використання різних форм виведення.

```
In[31]:= FullForm[Exp[x]^2/a]

Out[31]//FullForm=
        Times[Power[a, -1], Power[E, Times[2, x]]]

In[32]:= TreeForm[Exp[x]^2/a]

Out[32]//TreeForm=
```



```
In[33]:= PaddedForm[(x^3 + 2*x^2 + 3*x - 1) / (x - 1), 3]
```

Out[33]//PaddedForm=

$$\frac{-1 + \quad 3\,x + \quad 2\,x^{\ 2} + x^{\ 3}}{-1 + x}$$

```
In[34]:= PrecedenceForm[12*b/c, 5]
```

Out[34]= $\frac{12\,b}{c}$

```
In[35]:= SequenceForm[Exp[x]^2/a]
```

Out[35]= $\frac{e^{2x}}{a}$

```
In[36]:= TableForm[{{"x", "y"}, {1, 2}, {3, 4}, {5, 6}}]
```

Out[36]//TableForm=

x	y
1	2
3	4
5	6

```
In[37]:= Prefix[f[x^2]]
```

Out[37]= $f@x^2$

```
In[38]:= Unevaluated[Exp[x]^2/a]
```

Out[38]= Unevaluated$\left[\frac{\text{Exp}[x]^2}{a}\right]$

Лістинг 3

Елементи графічного інтерфейсу

Засоби Mathematica дозволяють створювати об'єкти графічного інтерфейсу користувача GUI (Graphic User Interface) ноутбуків, що робить останні набагато наочнішими і зручними в роботі.

Слайдери однокоординатні

Слайдери дозволяють плавно або дискретно змінювати значення певної змінної. У лістингу 4 показано свторення трьох слайдерів функцією Slider з різними варіантами значень параметрів.

Slider[x]- слайдер зі значенням x від 0 до 1.

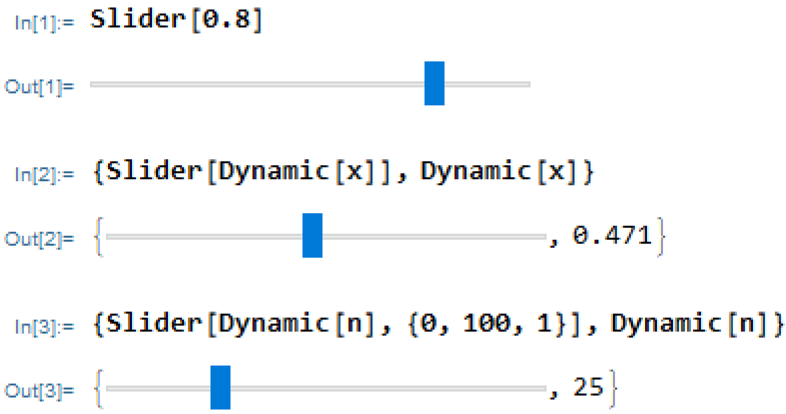
Slider[x,{xmin,xmax}]- слайдер зі значенням x від xmin до xmax.

Slider[x,{xmin,xmax,dx}]- слайдер зі значенням x від xmin до xmax з кроком dx.

Slider[x,{{e1,e2,…}}]- являє собою слайдер, в якому однакові інтервали відповідають послідовним налаштуванням.

Slider[x,{{{e1,w1},{e2,w2},...}}]- слайдер використовує інтервали відносної ширини wi для ei.

Типи слайдерів



Лістинг 4

У першому випадку (верхній слайдер) параметр (число 0,8) визначає положення його движка. При переміщенні движка буде повернено значення в діапазоні від 0 до 1. Другий слайдер (у центрі) дозволяє змінювати значення змінної x. Щоб зробити значення змінної динамічно доступним в усьому ноутбуці, змінна є динамічною, для її створення використовують функцію Dynamic [x]. Типово діапазон значень змінної становить від 0 до 1. Третій слайдер дозволяє розширити діапазон значень слайдера від 0 до 100 з кроком 1.

Двокоординатні слайдери

Для побудови поверхонь, які описують функціями двох змінниxвикористовують двокоординатні слайдери. Їх створюють функцією Slaider2D, лістинг 5.

Slider2D[{x,y}] – 2Д слайдер зі зміною значень x та y від 0 до 1.

Silder2D[Dynamic[pt]]- 2Д слайдер зі зміною значеньдинамічної змінної pt.

Slider2D[pt,{min,max}]-2Д слайдер зі зміною значень pt від min до max.

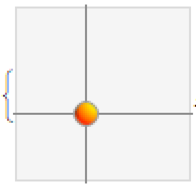
Slider2D[pt,{min,max,d}]-2Д слайдер зі зміною значень pt від min до max з кроком d.

Slider2D[pt,{{xmin,ymin},{xmax,ymax}}]- 2Д слайдер зі зміною значень pt від xmin до xmax по x координаті та від ymin до ymax по y координаті.

Slider2D[pt,{{xmin,ymin},{xmax,ymax},{dx,dy}}]- 2Д слайдер зі зміною значень pt від xmin до xmax по x координаті та від ymin до ymax по y координаті з кроками dx та dy відповідно.

Приклад створення двокоординатного слайдера представлений на рис. 2.

In[4]:= `{Slider2D[Dynamic[r]], Dynamic[r]}`

Out[4]= {  , {0.38, 0.37}}

In[5]:= `Dynamic[r]`

Out[5]= {0.38, 0.37}

Рис. 2. Двокоординатний слайдер

Движок такого слайдера може переміщатися мишею в будь-якому напрямку. Слайдер повертає два числових значення залежно від положення движка. Ці значення можна використовувати для обчислення функцій двох змінних і побудови графіків поверхонь, тривимірних фігур, параметрично заданих графіків і т.д.

Елемент графічного інтерфейсу користувача CheckBox

Нерідко обчислення необхідно проводити за умови встановлення деяких опцій, для чого зручно використовувати елемент CheckBox, який створюють функцією CheckBox[], лістинг 6.

Checkbox[*x*] –відображає чекбокс для x, який відображає ☑ коли x True, і ☐ коли x False.
Checkbox[Dynamic[*x*]]- відображає чекбокс для динамічної змінної x, який змінює значення x при натисканні на чекбох.
Checkbox[*x*,{*val*1,*val*2}]-відображає чекбокс для x, який відображає ☑ коли x val2, і ☐ коли x val1.
Checkbox[*x*,{*val*1,*val*2,*val*3,…}]- відображає чекбокс який перемикається між значеннями *val*i і відображає ▣для *val*i при i>2.

Лістинг 6

Приклади застосування функції CheckBox[] показано на рис. 3.

```
In[39]:= {Checkbox[False], Checkbox[True]}

Out[39]= {☐, ☑}

In[40]:= {Checkbox[1, {1, 2, 3}], Checkbox[2, {1, 2, 3}], Checkbox[3, {1, 2, 3}]}

Out[40]= {☐, ☑, ▣}

In[52]:= Checkbox[Dynamic[x], {1, 2}]

Out[52]= ☐

In[54]:= Dynamic[x]

Out[54]= 1
```

```
In[52]:= Checkbox[Dynamic[x], {1, 2}]
```

Out[52]= 

```
In[54]:= Dynamic[x]

Out[54]= 2
```

Рис. 3. Елемент графічного інтерфейсу користувача CheckBox

Локатор

Локатор – об'єкт, який має форму точки на графічному вікні та повертає її координати. Цей об'єкт представлений зафарбованим кружком в середині перехрестя, що має світлу крапку в середині. Локатор переміщують мишею. Для його створення використовують функцію, лістинг 7.

Locator[{*x*,*y*}] – відображає локатор в позиції {*x*,*y*}.

Locator[Dynamic[*pos*]]- відображає локатор з динамічною зміною значення pos, значення змінюється при зміні положення локатора.

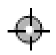
Locator[{*x*,*y*},*obj*] – відображає *obj* як об’єкт локатора.

Locator[{*x*,*y*},None]- відображає нічого візуального як об’єкт локатора.

Лістинг 7

Приклад створення локатора показано на рис. 4.

```
In[44]:= DynamicModule[{p = {0.5, 0.5}},
          {Graphics[Locator[Dynamic[p]], PlotRange → 2], Dynamic[p]}]
```

Out[44]= {  , {0.136364, 0.39899}}

```
In[33]:= Table[Graphics[Locator[{0, 0}, ImageSize → m]], {m, {10, 30, 50, 80}}]
```

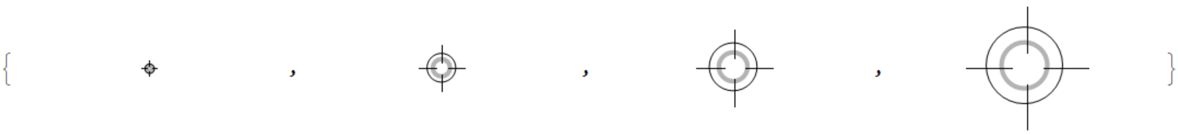
Out[33]= 

Рис. 4. Локатор

Локатори часто застосовують для побудови графіків по точкам.

Функції керування мишею

Іноді виникає необхідність у визначенні координат курсору миші, для чого використовують функцію MousePosition. Приклад її застосування показано у лістингу 8.

```
In[45]:= Dynamic[MousePosition[]]

Out[45]= {819, 554}
```

Лістинг 8

Координати миші виводятьу форматі цілих чисел. Додавання функції MousePosition в список параметрів функції Dynamic дозволяє вивести координати курсору миші у вигляді списку поточних координат.

Функція Opener [x] або Opener [Dynamic [x]] реагує на кожен клік миші по чорному трикутнику. Наприклад, команда {Opener [Dynamic [x]], Dynamic [x]} виводить в рядок виводу список:

```
In[28]:= {Opener[Dynamic[x]], Dynamic[x]}

Out[28]= {^, True}
```

Однак якщо клікнути по ньому мишею з'явиться список

```
In[28]:= {Opener[Dynamic[x]], Dynamic[x]}

Out[28]= {˅, False}
```

Функція Toggler[x]:

Toggler[x] – відображає циклічний перемекач, що перемикає значення x між True та False.

Toggler[Dynamic[x]] – відображає циклічний перемекач, що перемикає значення динамічної змінної x між True та False.

Toggler[x,{val1,val2,…}]– відображає циклічний перемекач, що перемикає значення x між послідовністю значень *val*i.

Toggler[x,{val1_>pict1,val2_>pict2,…}] -відображає циклічний перемекач, що перемикає значення x між послідовністю значень *val*i, де значення *val*i відображаються через *pict*i.

Toggler[x,vlist,dpict]- відображає dpict якщо x не дорівнює жодному з значень *val*i з списку vlist.

забезпечує можливість реакції на неодноразові натискання клавіши миші. Наприклад,

```
In[46]:= Toggler[1, {a, b, c, d}]


Out[46]= 1
```

спочатку виводиться 1. Однак, після натиснення кнопки миші на цифру 1 4 рази, виводяться символи a, b, c і d послідовно.

```
In[29]:= Toggler[3, {1 → "One", 2 → "Two", 3 → "Three"}]

Out[29]= One

In[30]:= Toggler[1, {1, 2, 3}, AutoAction → True]

Out[30]= 2
```

AutoAction->True перемикає значення при проведенні мишкою через результат.

Кнопка з написом

Нерідко деякі дії повинні виконуватися при натисканні кнопки з написом. Для створення такої кнопки використовують функцію Button [label, action], де label-назва кнопки, action-дія, що виконується після натискання кнопки. У списку її параметрів вказують рядок з ім'ям кнопки і вираз, який виконують за умови активізації кнопки мишею. Приклад застосування функції Button, з виведення факторіалу числа 10, зображено у лістингу 9.

```
In[48]:= Button["Click Here", Print[10!]]

Out[48]= Click Here

3628800

In[31]:= Table[With[{i = i}, Button[i, Print[i!]]], {i, 10}]

Out[31]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

1

2

6

24

120

720
```

Лістинг 9

Маніпулятор

Маніпулятор – об'єкт, схожий на слайдер, але має більші функціональні та візуальні можливості. Маніпулятор створюють за допомогою функції:

Manipulator[*x*]- відображає маніпулятор,який задає x від 0 до 1.

Manipulator[Dynamic[*x*]] - відображає маніпулятор,який задає динамічну змінну x від 0 до 1.

Manipulator[*x*,{*xmin*,*xmax*}]- відображає маніпулятор,який задає x від *xmin* до *xmax*.

Manipulator[*x*,{*xmin*,*xmax*,*dx*}]- відображає маніпулятор,який задає x від *xmin* до *xmax* з кроком *dx*.

Відмінною особливістю маніпулятора є характерна кнопка у вигляді сірого прямокутника зі знаком «+ » в ній. При активізації мишею цієї кнопки під слайдером маніпулятора з'являється панель з органами керування слайдером (див. рис. 5 , приклад зверху). За допомогою органів (кнопок) керування маніпулятора можна запустити його і забезпечити автоматичне переміщення движка слайдера, можна змінити напрямок і швидкість переміщення слайдера, здійснити його зупинку. За умови наближення слайдера до початкової та кінцевої точок у движка з'являється характерна тінь.

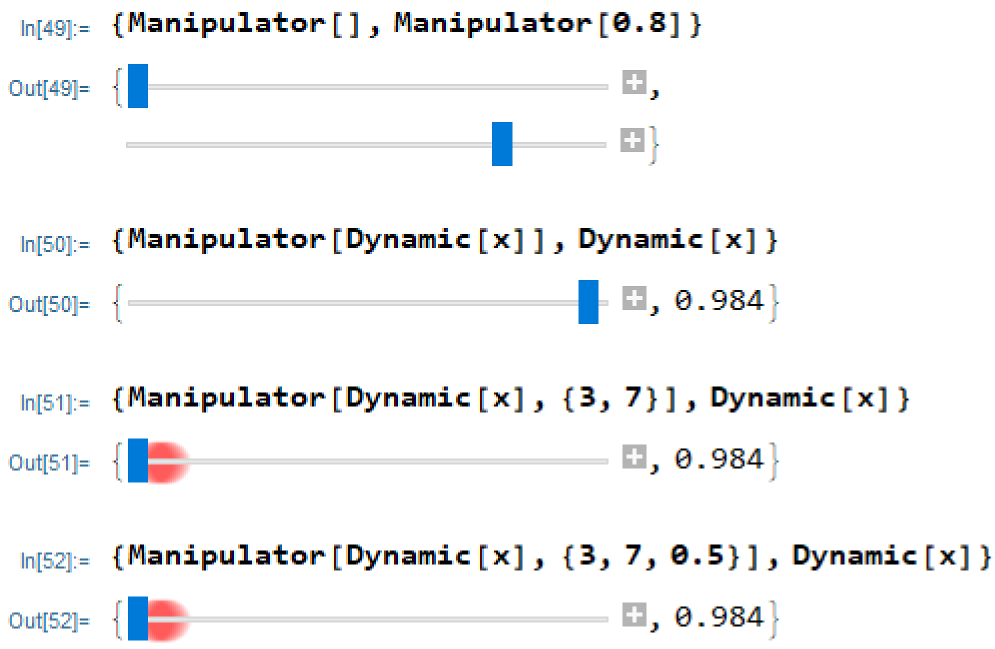


Рис. 5. Маніпулятор

Задавач кута повороту радіус-вектора

У деяких випадках, наприклад, при побудові графіків функцій в полярній системі координат, потрібен об'єкт, що задає кут повороту радіус-вектора. Такий об'єкт – задавач кута повороту – задається функцією DynamicModule[{x = RandomReal[{0,50}]}, {Experimental`AngularSlider[Dynamic@x], Dynamic@x}]. На рис. 6 представлений приклад застосування цієї функції.

```
In[13]:= DynamicModule[{θ = 0},
  Grid[
   {{DynamicModule[{x = RandomReal[{0, 50}]},
      {Experimental`AngularSlider[Dynamic@θ], Dynamic@θ}],
     Plot[Sin[x], {x, -10, 10},
      Epilog → {PointSize[Medium], Dynamic@Point[{θ, Sin[θ]}]}]},
    {ParametricPlot[{Cos[y], y}, {y, -10, 10}, AspectRatio → 2,
      Epilog → {PointSize[Medium], Dynamic@Point[{Cos[θ], θ}]}],
     Dynamic[θ]}}]]
```

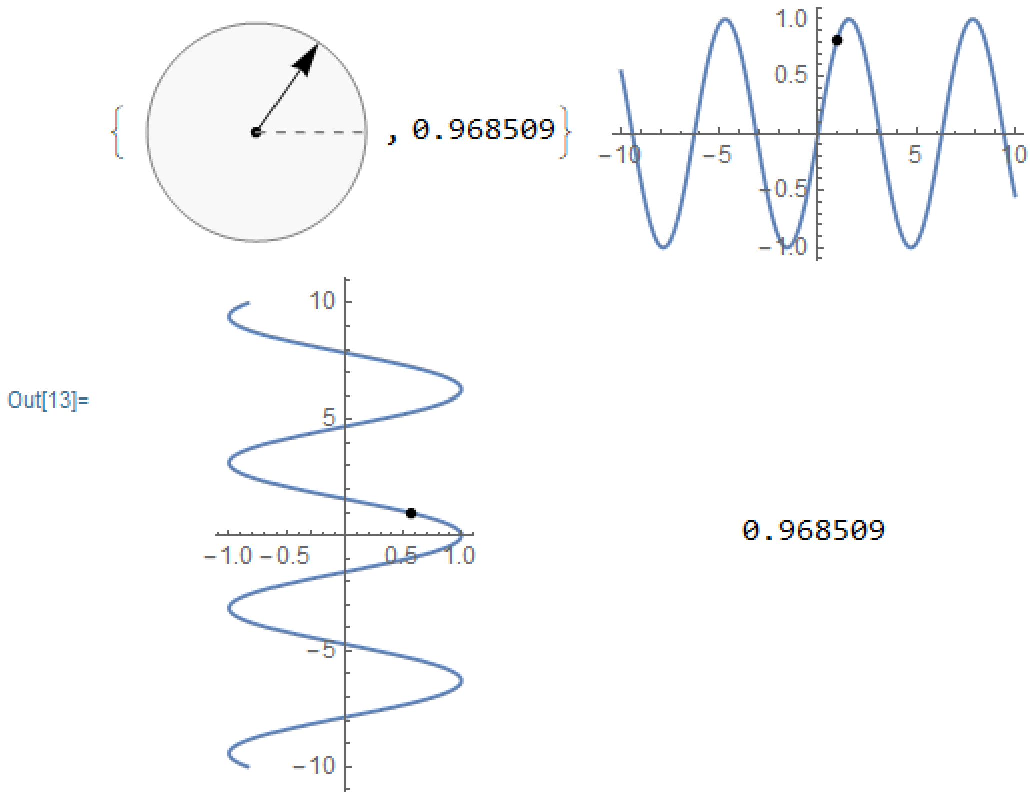


Рис. 6. Задавач кута повороту радіус-вектора

Програма з рисунку 6: DynamicModule[{$\theta = 0$}, Grid[{{DynamicModule[{$x =$ RandomReal[{0,50}]}, {Experimental`AngularSlider[Dynamic@θ], Dynamic@θ}], Plot[Si
{PointSize[Medium], Dynamic@Point[{θ, Sin[θ]}]}]}, {ParametricPlot[{Cos[y], y}, {y, −10
2, Epilog → {PointSize[Medium], Dynamic@Point[{Cos[θ], θ}]}], Dynamic[θ]}}]]

Задавач є графічним об'єктом у вигляді кола, всередині якого розміщено радіус-вектор. Він може обертатися в ту чи іншу сторону за допомогою миші, що веде до зміни задаваного задатчиком кута повороту. У прикладі, показаному на рис. 6, побудовані проекції кінця радіус-вектора на координатніїй осі. Ці проекції, як відомо, є синусоїдальні функції. Кут, що відповідає поточному положенню радіус-вектора, позначений крапками на графіках синусоїдальних функцій.

Випадаюче меню

Випадаюче меню – ще один широко застосовуваний об'єкт для побудови графічного інтерфейсу. Його створюють функцією ActionMenu[*name*,{*lbl*1:>*act*1,*lbl*2:>*act*2,…}]. Параметрами функції є рядок з написом на кнопці і список назв позицій меню і дій, виконуваних при активізації відповідних позицій меню. У прикладі, показаному на рис. 7, розраховують значення факторіалів 4!, 7! і 10!.

```
In[62]:= ActionMenu["Print Factorials",
          {"4!" :> Print[4!], "7!" :> Print["7!"], "10!" :> Print[10!]}]
```

Out[62]=

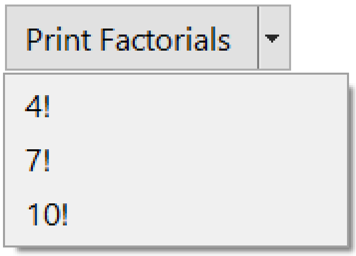


Рис. 7. Приклад створення меню

Панель введення даних

Панель введення виразів використовують для інтерактивного введення довільного виразу, наприклад, для побудови його графіка. Вона вводиться функцією Panel.

Panel[*expr*] – показує панель що містить expr.

Panel[*expr*,*title*] - показує панель з назвою title.

Panel[*expr*,*title*,*pos*] – розташовує назву title у позицію, що визначається через pos.

Panel[*expr*,{*title*1,*title*2,…},{*pos*1,…}]- розташовує назву *title*i у позицію *pos*i.

Panel[].

Приклад задання панелі введення виразу і побудови його графіка показаний на рис. 8.

```
In[69]:= Panel[DynamicModule[{f = Sin[x]},
           Column[{InputField[Dynamic[f]],
             Dynamic[Plot[f, {x, -5, 5}]]}]]]
```

Out[69]=

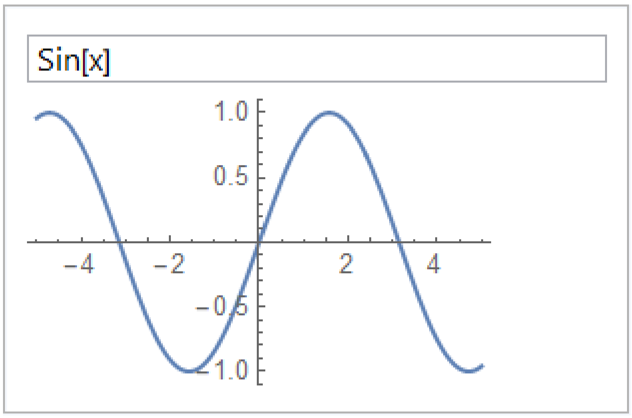


Рис. 8. Приклад створення панелі введення даних

Зверніть увагу на те, що в панелі введення за замовчуванням з'являється вираз (у нашому випадку Sin [x]), який як параметр функції DynamicModule всередині списку параметрів функції Panel. Цей вираз використовують для побудови графіка в області виведення.

Радіокнопки та меню налаштування

Функція RadioButton [x, val] створює так звану радіокнопку у вигляді кола. Результат виконання функції можна використовувати для програмного введення тієї або іншої дії. Приклади застосування радіокнопки дано на рис. 9.

```
In[80]:= {RadioButton[], RadioButton[False, True]}

Out[80]= {○, ○}

In[81]:= Table[RadioButton[Background → b],
          {b, {Pink, Green, Yellow, Orange}}]

Out[81]= {○, ○, ○, ○}

In[87]:= {RadioButton[Dynamic[x], 1, AutoAction → True],
          RadioButton[Dynamic[x], 2, AutoAction → True],
          Dynamic[x]}

Out[87]= {○, ○, x}

In[90]:= Table[RadioButton[Dynamic[x], a, Appearance → Dynamic[x]],
          {a, {Tiny, Small, Medium, Large}}]

Out[90]= {○, ○, ◉, ○}

In[93]:= Dynamic[x]

Out[93]= Medium
```

Рис. 9. Елемент RadioButton

Ще одна функція в ряді форматів запису задає побудову меню налаштувань має такі формати запису: 0

SetterBar[x,{$val1$,$val2$,…}]- відображає меню налаштувань x з налаштуваннями vali.

SetterBar[Dynamic[x],{$val1$,$val2$,…}]- відображає меню налаштувань для динамічно змінної x з налаштуваннями vali.

SetterBar[x,{$val1$->$lbl1$,$val2$->$lbl2$,…}]- відображає меню налаштувань x з налаштуваннями vali, де назва vali визначається як lbli .

Приклад її представлено на рис. 10.

```
In[97]:= Dynamic[Plot[Sign[Sin[n*x]], {x, 0, 2*Pi},
   PlotLabel → SetterBar[Dynamic[n], Range[10]]]]
```

Out[97]=

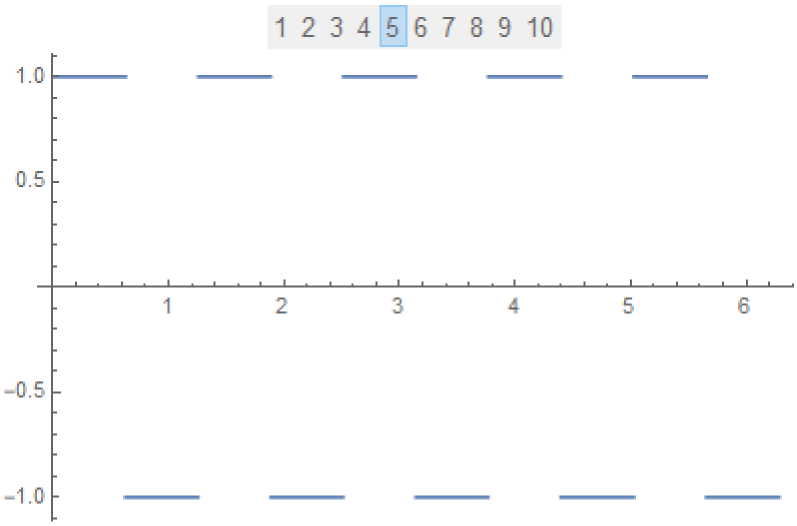


Рис. 10. Створення меню налаштувань функцією SetterBar[]

У цьому прикладі можливо, активізувавши ту чи іншу позицію меню з цифрою, задати частоту меандра.

Ще один варіант побудови меню налаштувань задає функція:

Setter[*x*, *val*]- відображає налаштування, при якому змінна x приймає значення val. Кнопка має назву val, при натисканні на неї змінна x приймає значення val.

Setter[Dynamic[*x*], *val*]- відображає налаштування, при якому динамічна змінна x приймає значення val. Кнопка має назву val, при натисканні на неї динамічна змінна x приймає значення val.

Setter[*x*, *val*, *label*]- відображає налаштування, при якому змінна x приймає значення val. Кнопка має назву label.

Setter[*x*, {*val*1, *val*2,…}, *label*]- відображає налаштування, при якому змінна x приймає значення *val*i. Кнопка має назву label.

. На рис. 11 показано приклад, в якому кнопки меню налаштувань забезпечують установку відповідного розміру графіка функції sin(x)^3.

```
In[98]:= {Setter[Dynamic[x], "Small"], Setter[Dynamic[x], "Medium"]}

Out[98]= {Small, Medium}

In[99]:= Dynamic[x]

Out[99]= Small

In[100]:= Plot[Sin[x]^3, {x, 0, 10}, ImageSize → Dynamic[x]]

Out[100]=
```

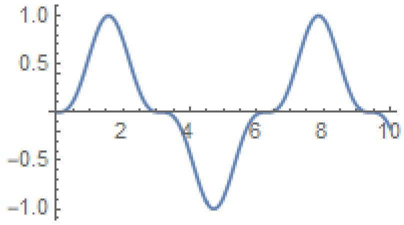


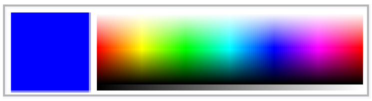
Рис. 11. Приклад використання функції Setter[]

Слайдер зміни кольору

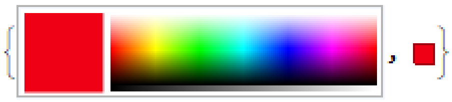
Для зміни кольору об'єктів графічного інтерфейсу зручно використовувати слайдер, котрий задають наступною функцією:ColorSlider[*color*], ColorSlider[Dynamic[*color*]], ColorSlider[].

Приклади застосування цієї функції показано на рис. 12.

```
In[105]:= ColorSlider[Blue]

Out[105]=
```



```
In[104]:= {ColorSlider[Dynamic[x]], Dynamic[x]}

Out[104]=
```



```
In[106]:= Dynamic[x]

Out[106]=
```

Рис.12. Слайдер зміни кольору

Колір першого квадрату – поточний колір. Ліворуч від нього – панель вибору кольору. Для вибору кольору досить встановити курсор миші на потрібний колір панелі і натиснути ліву клавішу миші. Колір контрольного квадрату стане аналогічним обраному.

Пусковий механізм

Функція Trigger імітує роботу пускового механізму. Вона має кілька форм запису:

Trigger[Dynamic[u]]- відображає пусковий механізм, при запуску якого змінна u змінюється від 0 до 1.

Trigger[Dynamic[u],{umin,umax}]- відображає пусковий механізм, при запуску якого змінна u змінюється від umin до umax.
Trigger[Dynamic[u],{umin,umax,du}]- відображає пусковий механізм, при запуску якого змінна u змінюється від umin до umax з кроком du.
Trigger[Dynamic[u],{umin,umax},ups]- відображає пусковий механізм, при запуску якого змінна u змінюється від umin до umax з кроком ups кожної секунди.

Результатом виконання функції, показаним на рис. 13, є панель керування пусковим «механізмом».

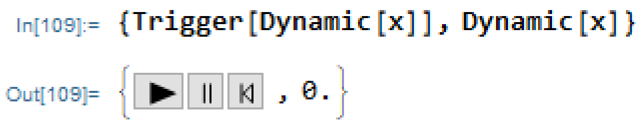


Рис. 13. Пусковий механізм

При натисканні на кнопку пуску (великий трикутник) значення динамічної змінної x починає змінюватися від 0 до 1 протягом декількох секунд. Потім ця зміна припиняється. На панелі є також кнопки зупинки змінних і повернення до 0 (Reset).

Функції позначення додаткової інформації на графіках

Функцію ClickPane[image,func], ClickPane[image,{{xmin,ymin},{xmax,ymax}},func] використовують для позначення точки на малюнку image із застосоуванням виразу func. Найчастіше функцію ClickPane використовують для вибору курсором миші деякого місця малюнка, в яке необхідно клацанням миші помістити якийсь графічний об'єкт, наприклад точку, вістря стрілки, коло і т.д. На рис. 14 показано приклад застосування функції для позначення екстремуму синусоїдальної функції за допомогою стрілки, що виходить від напису «Екстремум».
DynamicModule[{pt={Pi/2,3}},ClickPane[Plot[3Sin[x],{x,-18,6},Epilog->{Dynamic@Arrow[{{2,5},pt}],Text["Екстремум",{4,5}]},PlotRange->6],(With[{x=(2Round[-1/2+2#[[1]]/Pi]+1)Pi/2},pt={x,3Sin[x]}])&]] – програма з прикладу.

```
In[111]:= DynamicModule[{pt = {Pi / 2, 3}},
  ClickPane[Plot[3 Sin[x], {x, -18, 6},
    Epilog → {Dynamic@Arrow[{{2, 5}, pt}], Text["Екстремум", {4, 5}]},
    PlotRange → 6],
   (With[{x = (2 Round[-1 / 2 + 2 #[[1]] / Pi] + 1) Pi / 2},
     pt = {x, 3 Sin[x]}]) &]]
```

Out[111]=

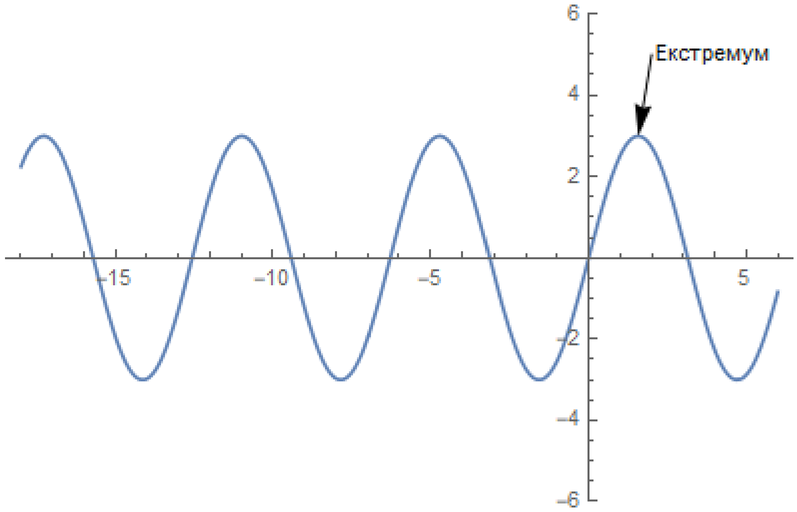


Рис. 14. Приклад використання функції ClickPane[]

Фунція Tooltip[*expr*, *label*] виводить вираз expr і заміняє його написом label, якщо курсор миші встановлено на вираз expr. На рис. 15 показано найпростішу реалізацию цієї функції.

```
In[112]:= Tooltip[2 + 3, "Це сума 2+3"]

Out[112]= 5
```

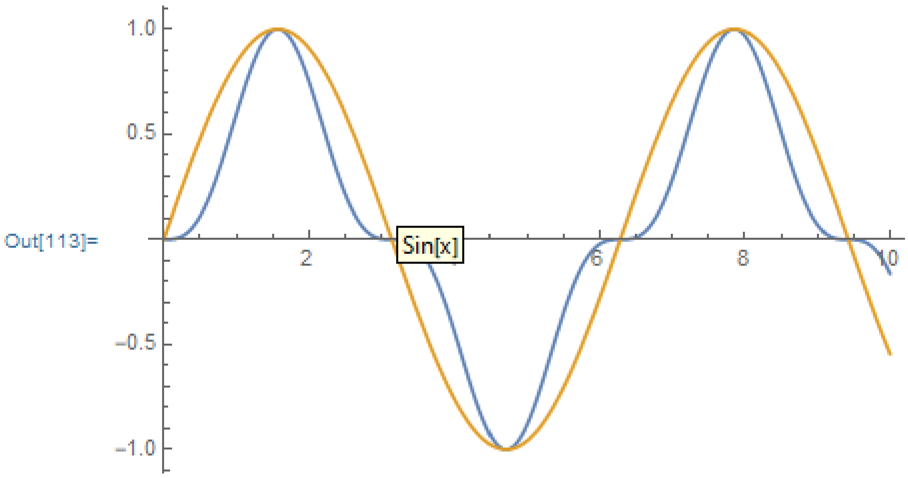
Це сума 2+3

Рис. 15. Приклад використання функції Tooltip[]

Якщо на результат, виведений у рядку навести курсор миші, буде показано текстовий напис «Це сума 2 +3», яка була задана як параметр label.

В іншому прикладі, рис.16, показано застосування цієї функції для розпізнання однієї з двох кривих, побудованих одним кольором.

```
In[113]:= Plot[Tooltip[{Sin[x]^3, Sin[x]}], {x, 0, 10}]
```




```
In[113]:= Plot[Tooltip[{Sin[x]^3, Sin[x]}], {x, 0, 10}]
```

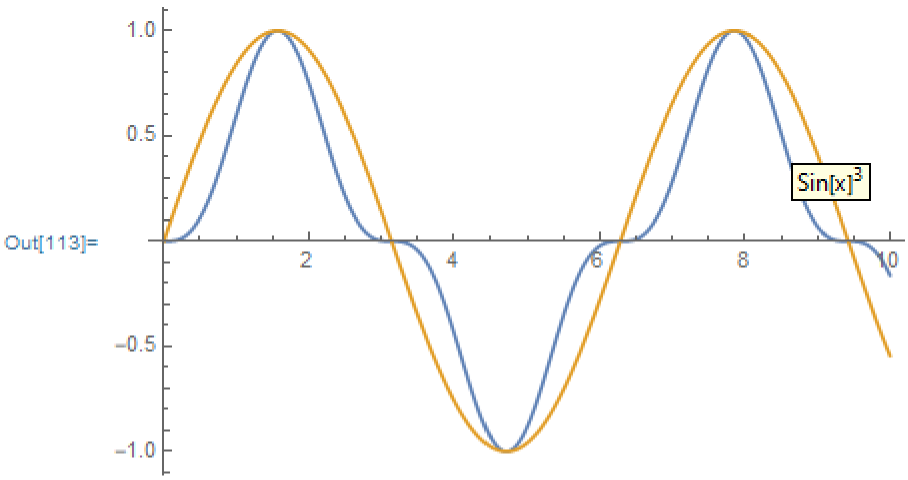


Рис. 16. Розпізнавання графіка функцією Tooltip[]

Для розпізнавання графіка функцією Tooltip[] необхідно навести курсор миші на одну з кривих.

На рис. 17 показано приклад застосування функції Tooltip[] для визначення точного значення ординати точкового графіка синусоїди під час наведення на потрібну точку курсору миші.

```
In[114]:= ListPlot[Table[Tooltip[N[Sin[i/2]]], {i, 20}]]
```

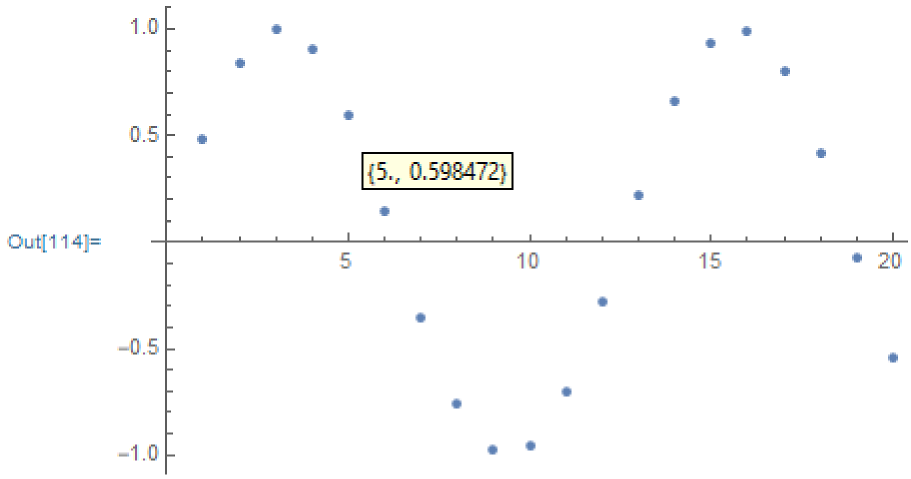


Рис. 17. Визначення значення ординати точкового графіка функцією Tooltip[]

Повідомлення, активізовані курсором

Функція PopupWindow[] забезпечує контроль над наведенням курсора мишіна поставлений в ній об'єкт. Якщо курсор наведений на об'єкт, то він модифікується і з'являється панель з заданим повідомленням. Наприклад, на рис. 18 об'єктом є коло, що в момент наведення на нього курсору миші змінює забарвлення. Потім після появи панелі з написом «Це диск» коло відновлює забарвлення.

```
In[115]:= PopupWindow[Graphics[Disk[], ImageSize → 50], "Це диск"]
```

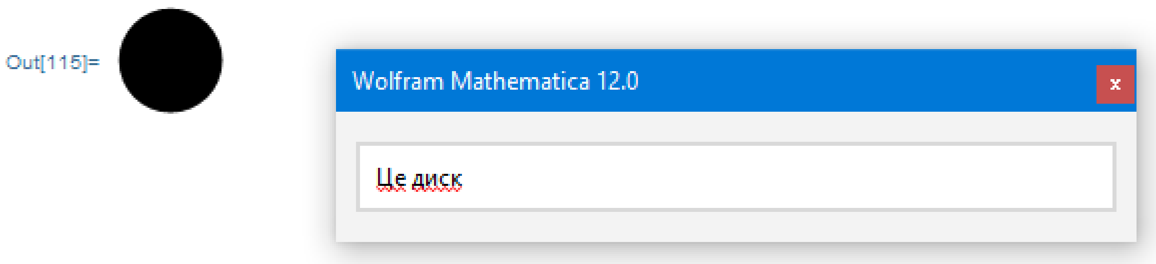


Рис. 18. Повідомлення, активізовані курсором

Виведення меню і вибір його позиції

Функція MenuView[] забезпечує створення меню з випадаючими позиціями під номерами, які відповідають значенню змінної n. У прикладі рис. 19 обчислюють функцію tan(nx) для різних n.

Виведення меню з вкладеннями та навігація між ними

Функція TabView[] виводить меню з вкладеннями, перехід між якими здійснюють натисканням лівої кнопки миші. Приклад застосування цієї функції показано на рис. 20. У даному випадку у вікні об'єкта будують малюнок, який створено клітинним автоматом за допомогою функції CellularAutomaticc[] з різним значенням параметра r (rule). Можливі значення r вибирають зі списку активізацією відповідної вкладинки мишею.

In[116]:= `MenuView[Table[Plot[Tan[n*x], {x, -4, 4}], {n, 5}]]`

Out[116]=

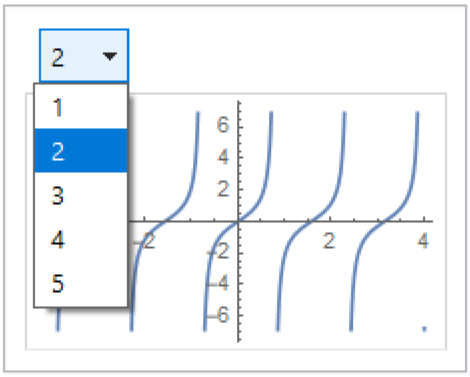


Рис. 19. Приклад використання функції MenuView[]

In[117]:=
```
TabView[
 Table[Row[{"rule", r}] →
   ArrayPlot[CellularAutomaton[r, {{1}, 0}, 30]],
  {r, {30, 50, 90, 150}}]]
```

Out[117]=

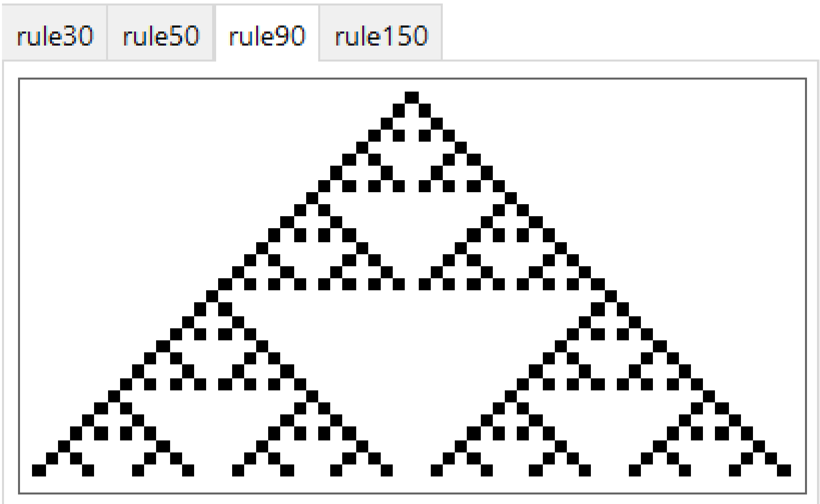


Рис. 20. Приклад використання функції TabView[]

Слайд- меню

Слайд-меню використовують для створення презентацій. Для створення слайд-меню використовують функцію SlideView[]. Роботу з функцією показано на рис. 21. При активізації кнопок із зображеннями трикутника можна вибирати символ зі списку. Перемикання може йти в будь-яку сторону, а також відразу на початок або в кінець списку.

In[118]:= `SlideView[{a, b, c, d}]`

Out[118]=

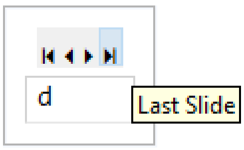


Рис. 21. Приклад використання функції SlaidMenu[]

Створення вікон інтерактивного діалогу з користувачем

У програмі Mathematica передбачено можливість створення діалогових вікон на основі пакету розширення GUIKit . На рис. 22 показано виклик цього пакету і вікна для демонстрації арифметичних операцій з двома числами що вводяться. Для створення навіть такого простого прикладу потрібна програма, що містить близько шестидесяти рядків . Спочатку потрібно підключити розширення `GUIKit`` використовуючи функцію Needs["GUIKit`"].

```
In[119]:= Needs["GUIKit`"];

In[120]:= GUIRunModal["Wolfram/Example/Calculator"]
```

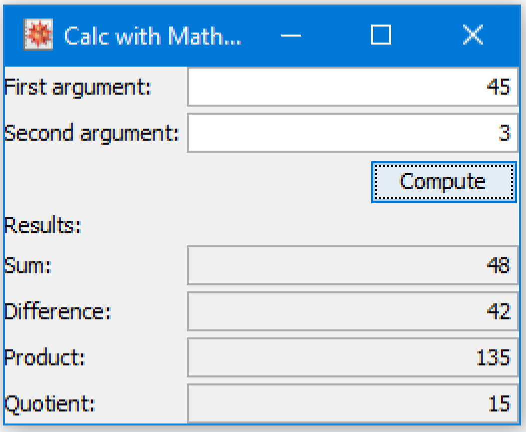


Рис. 21. Приклад використання GUIKit

Хід роботи

Написати програму, що демонструє можливості роботи з елементами графічного інтерфейсу згідно з варіантом, вказаним у табл. 1.

Таблиця 1

Вар.	Елементи графічного інтерфейсу
1,11	Слайдер однокоординатний, випадаюче меню, функція PopupWindow[]
2,12	Двокоординатний слайдер, панель введення даних, функція MenuView[]
3,13	Елемент CheckBox, функція RadioButton[], функція TabView[]
4,14	Локатор, функція SetterBar[],слайд-меню
5,15	Функія MousePosition[], функція Setter[], діалогове вікно користувача
6,16	Функція Input[], функція Print[],локатор
7,17	Функція Opener[], слайдер зміни кольору, елемент CheckBox
8,18	Кнопка, пусковий механізм, двокоординатний слайдер
9,19	Маніпулятор, функція ClickPane[],функція MousePosition[]
10,20	Слайдер однокоординатний, функція Tooltip[], кнопка

Контрольні запитання

1. Перелічіть функції введення-виведення системи Mathematica.
2. Назвіть формати виведення даних, які підтримує система Mathematica.
3. Перелічіть типи слайдерів і назвіть їх функціональне призначення.
4. Назвіть елементи графічного інтерфейсу, призначені для вибору однієї з альтернатив.
5. Назвіть функції роботи з мишею.
6. Перелічіть функції для створення позначок на графіках.
7. Назвіть функції для вибору кольору.
8. Вкажіть функції, які використовують для створення меню.
9. Опишіть принцип використання функції SlaidMenu[].
10. Пояснять функціональне призначення функції пускового механізму.

Література

1. В.З. Аладьев, Д.С. Гринь "Расширение функциональной среды системы Mathematica" / Монография / Херсон: Олди-Плюс, 2012, 552 с.
2. В.З. Аладьев, В.К. Бойко, Е.А. Ровба "Программирование в пакетах Maple и Mathematica: Сравнительный аспект" / Монография / Гродно: Гродненский Госуниверситет, 2011, 517 с.
3. Mathematica 5/6/7. Полное руководство / Дьяконов В. П. - ДМК Пресс – 2009, 624 стр.
4. Mathematica. Практический курс с примерами решения прикладных задач / Васильев А. Н. - КОРОНА-Век - 2008.
5. Mathematica / А. Половко - БХВ-Петербург – 2007 - 368 стр.
6. Введение в систему символьных, графических и численных вычислений "Математика-5" / Е. Воробьев - "Диалог-МИФИ" - 2005 - 368 стр.
7. Differential Equations with Mathematica, Third Edition / Brian R. Hunt, Ronald L. Lipsman, John E. Osborn, Donald A. Outing, Jonathan Rosenberg - 2009 John Wiley & Sons, 271 pp.
8. A Physicist's Guide to Mathematica, Second Edition / Patrick T. Tam – 2008 Academic Pres, 728 pp.
9. Computer Solutions in Physics: With Applications in Astrophysics, Biophysics, Differential Equations, and Engineering / Steve VanWyk - World Scientific 2008 - 282 pp.
10. Mathematica by Example, Fourth Edition / Martha L. Abell, James P. Braselton Publisher: Academic Press 2008 - 576 pp.
11. Mathematica DeMYSTiFied / Jim Hoste - McGraw-Hill Professional 2008 - 320 pp.
12. Mathematica Navigator: Mathematics, Statistics and Graphics, Third Edition / Heikki Ruskeepaa Academic Press 2009 - 1136 pp.